Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Semiannual Technical Report
covering the period
January 15, 1972 - August 30, 1972

Form Approved Budget Bureau No. 22-R0293

Sponsored by
Advanced Research Project Agency
ARPA Order No. 1997

ARPA ORDER Number:
1997/12-02-71

Program Code Number:
80230

Contractor:
Massachusetts Institute of Technology
Cambridge, Mass. 02139

Effective Date of Contract:
January 15, 1972

Contract Expiration Date:
January 14, 1973

Amount of Contract:
$96,377.00

Contract Number:
N00014-67-A-0204-0064

Principal Investigator:
Alan V. Oppenheim
617-253-4177

Scientific Officer:
Director, Information Systems
Mathematical and Information Sciences Division
Office of Naval Research
Department of the Navy
800 North Quincy Street
Arlington, Virginia 22217

Short Title of Work:
Speech and Picture Processing

DISCLAIMER NOTICE

THIS DOCUMENT IS THE BEST QUALITY AVAILABLE.

COPY FURNISHED CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*
Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP
None

3. REPORT TITLE

Semiannual Technical Report

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
Research Progress Report for period ending August 30, 1972

5. AUTHOR(S) *(First name, middle initial, last name)*

A. V. Oppenheim

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
August 30, 1972	8	6

8a. CONTRACT OR GRANT NO.
ONR Contract N00014-67-A-0204-0064

b. PROJECT NO.
ARPA Order No: 1997/12-71

c.
Program Code No: 80230

d.

9a. ORIGINATOR'S REPORT NUMBER(S)
None

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10. DISTRIBUTION STATEMENT

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY
Advanced Research Projects Agency
of the Department of Defense

13. ABSTRACT

During the first half of the present contract year, the program initiated or continued the following studies: Speech analysis by linear prediction, reconstruction of multidimensional signals from projections, applications of digital frequency warping, and development of a digital speech synthesizer. A review was also made of design and synthesis of digital filters within the constraints of finite register length. These projects are summarized and reprints of available publications are appended.

DD FORM 1 NOV 65 1473 (REPLACES

S/N 0102-014-6600

2

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Speech analysis						
Digital filters						
Linear prediction						
Multidimensional signals						

Semiannual Technical Report

ONR Contract N00014-67-A-0204-0064

covering the period

January 15, 1972 - August 30, 1972

submitted by

A. V. Oppenheim

August 30, 1972

Projects Studied Under the Contract

During the first half of the present contract year, the program initiated or continued the following studies: Speech analysis by linear prediction, reconstruction of multidimensional signals from projections, applications of digital frequency warping, and development of a digital speech synthesizer. A review was also made of design and synthesis of digital filters within the constraints of finite register length. These projects are summarized in the following pages. Reprints of available publications are appended.

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Advanced Research Projects Agency or the U. S. Government.

1. Speech Analysis by Linear Prediction

Recently, the analysis of speech by means of a technique referred to as linear prediction has received considerable attention. This technique is directed toward modeling a sequence as the output of an all-pole digital filter. The work carried out under this research contract is directed toward applying the techniques of linear prediction to the extraction of parameters for automatic speech recognition. Furthermore, we are at present investigating a number of alternative formulations of the technique together with some of the theoretical limitations.

A portion of the work relating to the extraction of parameters for speech recognition is being carried out on the fast digital processor facility at Lincoln Laboratory. The system implemented is capable of performing the analysis in real time, and effort is now directed toward the evaluation of its performance. Thus far the results are encouraging, indicating that with the use of linear prediction good speech parameter data can be extracted. This aspect of the work is reported in detail by V. W. Zue, Quarterly Progress Report No. 105, Research Laboratory of Electronics, M.I.T., April 1972, pp. 133-142.

A second aspect of this work relates to some possible theoretical shortcomings of the technique. In particular, it has been shown that in certain situations it is possible for the linear prediction technique to generate approximations to the speech spectrum with large errors. These results, and a comparison of a number of formulations of the linear prediction technique, have been summarized by M. R. Portnoff, V. W. Zue and A. V. Oppenheim in Quarterly Progress Report No. 106, Research Laboratory of Electronics, M.I.T., July 1972, pp. 141-150.

Based on the results obtained so far, the linear prediction technique for speech analysis appears to be extremely promising, but it has some theoretical pitfalls. Research under this contract on these problems is continuing.

2. Reconstruction of Multidimensional Signals from Projection

In a variety of contexts, projections of multidimensional signals are available, and a reconstruction of the original signal is desired. The basis

for the technique is that the Fourier transform of the projeetion of a signal ean be shown to be a slice through the Fourier transform of the original signal. The problem of reconstrueting multidimensional signals from their projections is eneountered naturally in a variety of eontexts ineluding x-ray photographs and electron mierography. The purpose of the present research is to formulate the reconstruction problem entirely in terms of diserete signals. We approaehed the problem from this point of view, and have obtained some partieularly interesting results. It has been shown that it is possible under relatively mild assumptions to eompletely define a multidimensional signal in terms of a single one-dimensional projection. In addition, a number of algorithms have been devised for reconstrueting multidimensional signals from a small number of projections. These results have potential applieation to a number of problems. The possibility is suggested that bandwidth eompression of multidimensional signals ean be accomplished by coding in terms of projeetion. Furthermore, the use of projeetions to describe a multidimensional signal appears to hold some promise for earrying out the design of multidimensional filters. The theoretical basis for these algorithms and some examples are deseribed by R. M. Mersereau in Quarterly Progress Report No. 105, April 1972, Research Laboratory of Electronies, M.I.T., pp. 169-183.

3. Applieations of Digital Frequency Warping

Recently, a teehnique was proposed for proeessing a signal in such a way as to implement a nonlinear distortion in the frequeney axis. We are investigating applieation of this digital frequeney warping to a number of problems, partieularly the implementation of unequal resolution speetrum analysis. We have been investigating the approximation to eonstant percentage bandwidth that ean be achieved using this teehnique in conjunction with the fast Fourier transform algorithm. We have also been investigating the application of this teehnique to Vernier speetrum analysis. If small errors are allowable, it is possible to use digital frequency warping for approximately eonstant pereentage bandwidth frequency analysis, and it appears that the teehnique will be applieable to

Vernier spectrum analysis. The need for both unequal resolution and Vernier spectral analysis arises in a variety of contexts, including radar and sonar processing. We anticipate that the present research will have application to those problems. The details of the technique and some preliminary results are described by A. V. Oppenheim and D. H. Johnson in "Discrete Representation of Signals," Proc. IEEE 60, 681-691 (1972).

4. Development of a Digital Speech Synthesizer

We are working on the design and fabrication of a small, fast, inexpensive digital processor to be used primarily for speech synthesis, but with application to more general signal-processing tasks. At present, a detailed design of this processor has been made and hardware construction will begin shortly. A central component of the processor is a new high-speed multiplier that has been designed with partial support from this contract. This multiplier is described by J. Allen and E. R. Jensen in Quarterly Progress Report No. 105, Research Laboratory of Electronics, M.I.T., April 1972, pp. 147-152.

The synthesizer, when completed, will be connected to the PDP-9 computer. It will operate as a real-time synthesizer and will play an important role in a number of future research projects including speech bandwidth compression, speech analysis, and speech synthesis by rule.

5. Design and Synthesis of Digital Filters within the Constraints of Finite Register Length

Although in most cases the design of design of digital filters is carried out without regard to finite register length, they must be implemented with finite register length. The effects of finite register length manifest themselves in a number of ways, including parameter inaccuracies and truncation or rounding after arithmetic operations. Recently, a review of these effects was carried out. These results are discussed in detail in an invited paper by A. Oppenheim and C. Weinstein entitled "Effects of finite register length in digital filtering and the fast Fourier transform" published in Proc. IEEE 60 (1972). During the coming year, new research on these effects and the synthesis of digital filters will be pursued.

Publications

J. Allen and E. R. Jensen, "A New High-Speed Multiplier Design," Quarterly Progress Report No. 105, Research Laboratory of Electronics, M.I.T., April 1972, pp. 147-152.

R. M. Mersereau, "Recent Advances in the Theory of Reconstructing Multidimensional Signals from Projections," Quarterly Progress Report No. 105, Research Laboratory of Electronics, April 1972, pp. 169-183.

V. W. Zue, "Speech Analysis by Linear Prediction," Quarterly Progress Report No. 105, Research Laboratory of Electronics, M.I.T., April 1972, pp. 133-142.

A. V. Oppenheim and D. H. Johnson, "Discrete Representation of Signals," Proc. IEEE 60, 681-691 (1972).

M. R. Portnoff, V. W. Zue, and A. V. Oppenheim, "Some Considerations in the use of Linear Prediction for Speech Analysis," Quarterly Progress Report No. 106, July 1972, pp. 141-150.

A. V. Oppenheim, C. Weinstein, "Effects of Finite Register Length in Digital Filtering and the Fast Fourier Transform," Proc. IEEE 60 (1972).

C. Braccini and A. Oppenheim, "Applications of Digital Frequency Warping to Unequal Resolution and Vernier Spectrum Analysis," To be presented at Florence Seminar on Digital Filtering, Sept. 1972.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
RESEARCH LABORATORY OF ELECTRONICS
Cambridge, Massachusetts 02139

Reprinted from
Quarterly Progress Report No. 105, April 15, 1972

14/7

XI. COGNITIVE INFORMATION PROCESSING*

Academic and Research Staff

Prof. M. Eden
Prof. J. Allen
Prof. B. A. Blesser
Prof. T. S. Huang
Prof. F. F. Lee
Prof. S. J. Mason
Prof. W. F. Schreiber
Prof. D. E. Troxel
Prof. I. T. Young
Dr. R. R. Archer
Dr. G. H. Granlund
Dr. E. G. Guttmann
Dr. L. P. A. Henckels
Dr. K. R. Ingham
Dr. J. I. Makhoul
Dr. D. M. Ozonoff
Dr. O. J. Tretiak
F. X. Carroll
Deborah A. Finkel
E. R. Jensen
J. W. Modestino
J. S. Ventura

Graduate Students

B. S. Barbay
J. Bonn
W. T. Borroz
J. E. Bowie
B. E. Boyle
P. Coueignoux
J. R. Ellis, Jr.
I. S. Englander
A. M. Fakhr
A. E. Filip
S. G. Finn
W. B. Grossman
D. W. Hartman
M. Hubelbank
F. H. Ives
J. W. Klovstad
H. S. Magnuski
L. A. Marks
P. L. Miller
B. Minkow
O. R. Mitchell, Jr.
J. A. Myers
D. O'Shaughnessy
R. A. Piankian
G. Poonen
R. S. Putnam
R. San Antonio
R. J. Shillman
D. G. Sitler
J. R. Sloan
A. A. Smith
R. D. Solomon
M. Sporer
J. H. Stahler
W. D. Stratton
H-m. D. Toong
K. P. Wacks
Y. D. Willems
H. M. Wolfson
K. H. Yung

A. NEW HIGH-SPEED MULTIPLIER DESIGN

With the advent of MSI and LSI integrated circuit technology, there is no doubt that digital multipliers of very high speed can be achieved, once it is agreed what should be incorporated in these chips.[1] In the meantime, we can achieve a very fast design by simultaneously exploiting the mathematical structure of binary two's-complement multiplication and existing MSI circuits which can be adapted in a natural way to the structure of this task. Accordingly, we shall show that the expression for binary multiplication can be rewritten to suggest use of the 74181 Arithmetic Logic Unit (ALU) in a straightforward way that achieves high speed, simple layout, and very little logic external to the ALU array.

In order to display the desired structure of multiplication, we shall consider the multiplication of two 4-bit two's-complement numbers. Let each such number be represented as

*This work was supported by the National Institutes of Health (Grant 5 PO1 GM14940-05), and by the Joint Services Electronics Programs (U.S. Army, U.S. Navy, and U.S. Air Force) under Contract DAAB07-71-C-0300).

$$Z \equiv z_3 z_2 z_1 z_0 = -2^3 z_3 + 2^2 z_2 + 2^1 z_1 + 2^0 z_0,$$

where each z_i is either 1 or 0, so that the product of two such numbers is

$$\begin{aligned} XY = {} & 2^6 x_3 y_3 - 2^5 (x_3 y_2 + x_2 y_3) + 2^4 (-x_3 y_1 + x_2 y_2 - x_1 y_3) \\ & + 2^3 (-x_3 y_0 + x_2 y_1 + x_1 y_2 - x_0 y_3) \\ & + 2^2 (x_2 y_0 + x_1 y_1 + x_0 y_2) + 2^1 (x_1 y_0 + x_0 y_1) + 2^0 x_0 y_0. \end{aligned}$$

This sum is commonly arranged in an array in which each column contains factors of like powers of 2, as in Fig. XI-1. The factors can be further rearranged, as

$$\begin{array}{r} -x_3y_0 + x_2y_0 + x_1y_0 + x_0y_0 \\ -x_3y_1 + x_2y_1 + x_1y_1 + x_0y_1 \\ -x_3y_2 + x_2y_2 + x_1y_2 + x_0y_2 \\ +x_3y_3 - x_2y_3 - x_1y_3 - x_0y_3 \end{array}$$

Fig. XI-1. Array representation of two's -complement multiplication.

$$\begin{array}{l} + \left\{ \begin{array}{llll} & & x_2y_0 & x_1y_0 \quad x_0y_0 \\ & x_2y_1 & x_1y_1 & x_0y_1 \\ x_2y_2 & x_1y_2 & x_0y_2 & \\ x_3y_3 & & & \end{array} \right. \\ \hline - \left\{ \begin{array}{lll} x_3y_2 & x_3y_1 & x_3y_3 \\ x_2y_3 & x_1y_3 & x_0y_3 \end{array} \right. \end{array}$$

Fig. XI-2. Multiplication array, grouped in positive and negative terms.

shown in Fig. XI-2, in order to group positive and negative terms. At this stage, each row has a constant factor for each term such as y_0 for the top row, and these can be factored out as bits that control the conditional inclusion of a given row in the final sum. Thus, in Fig. XI-3, the top row ($x_2\ x_1\ x_0$) will be added into the sum just in case $y_0 = 1$, and similarly for the other rows. Figure XI-3 also shows 6 conditional terms to be summed, but one of these, x_3y_3, affects only the most significant bit position. If this term is included in any other row (say, row 6), the only change

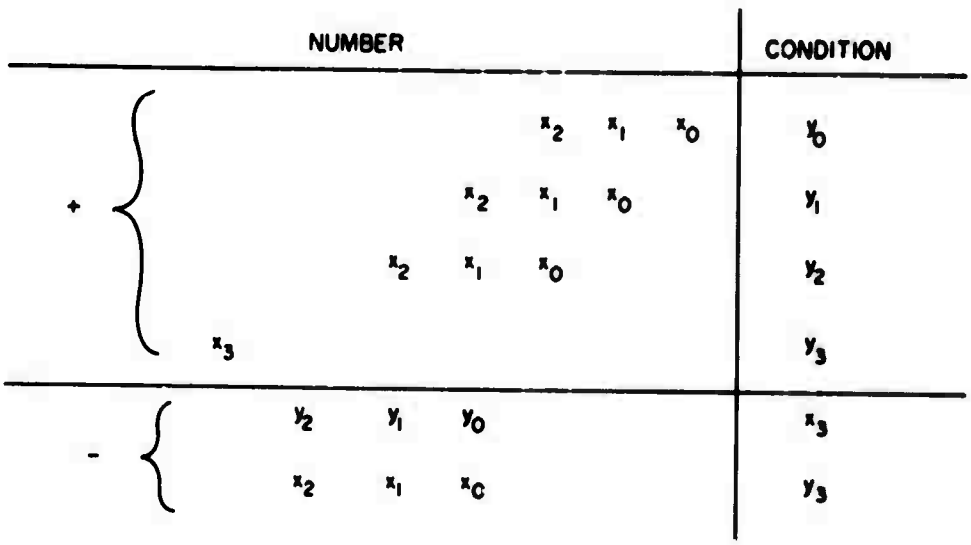

Fig. XI-3. Multiplication array, grouped by rows and their respective control bits.

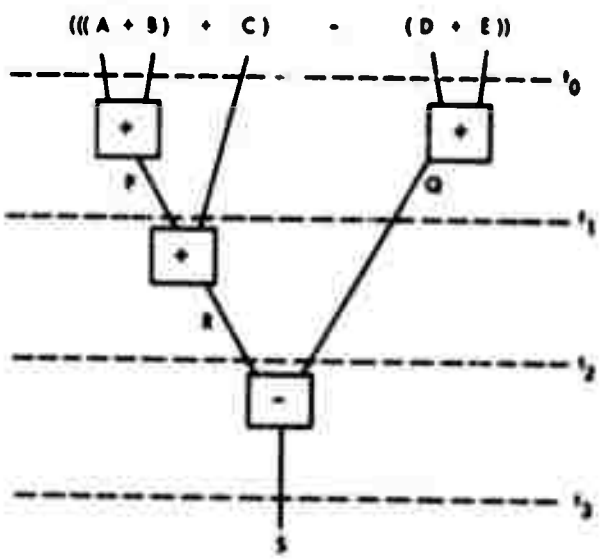

Fig. XI-4. Illustrating the parallel nature of the multiplication task.

in the final result might be a carry out of the most significant bit. When sign extension is not desired, it is thus possible to incorporate the x_3y_3 term in row 6, and we are left with just 5 rows to sum. It is this representation of the product as a sum of conditional terms that can be exploited by the ALU design.

Having rewritten the product as a sum of terms, each conditioned on a control bit, the next step is to minimize the sum-and-carry delays by exploiting the inherent parallelism of the array. Referring to the five rows as A through E, Fig. XI-4 shows how a tree structure permits simultaneous sums to be computed, rather than performing each indicated sum in serial, left-to-right order. In Fig. XI-4, each box denotes an ALU adder, and we assume that each add is completed in Δ seconds. During the first Δ seconds two adds are completed, followed by one add in each of the two succeeding intervals. There are two advantages in this scheme. First, the total delay would be 4Δ seconds if the adds were done serially, but when the parallelism is utilized, only 3Δ seconds of delay result. More generally, for larger sized numbers N bits long, a similar binary tree would lead to $(\log_2 N)\Delta$ seconds delay, whereas a serial procedure would require $(N-1)\Delta$ seconds delay. For N = 16, the saving is $\Delta(15-4) = 11\Delta$, a very substantial figure. When N is not a power of 2, some branches of the full binary tree are pruned, but the saving in time because of parallelism is still obtained. The second advantage is that the binary tree arrangement can be implemented in a straightforward and natural way by using the 74181 ALU, 24-pin MSI package.

The 74181 ALU, shown in Fig. XI-5, operates on two 4-bit inputs in a manner prescribed by the four control bits, S_0 through S_3, and the Mode Control bit M, to produce a single 4-bit output. As shown in Fig. XI-6, all of the needed control functions can be realized by appropriate use of S_0 through S_3, M, and $\overline{C}_0$, the last being the input carry to the least significant bit. Note that only the double conditional sum, (A if z) + (B if y), requires extra circuitry to translate the condition bits (z and y) into ALU controls, but that this circuitry is very simple, containing only an XOR gate and an inverter.

Figure XI-7 shows the complete design for a 4×4 multiplier, in which the control circuitry is shown in detail. Depending on the size of the multiplier desired, extra time savings may be realized by appropriate partitioning of the array and insertion of carries, but the basic details remain the same. The authors have designed 16×16 and 16×24 arrays, which illustrate further refinements. These designs are available to the interested reader.

A further advantage of the ALU is its wide availability. Originally, it was introduced in TTL, but Schottky TTL and MECL 10,000 versions are now available. Worst-case multiplication times will depend on which one of these packages is used, but a 16×16 design should yield a completion time of 95-100 ns

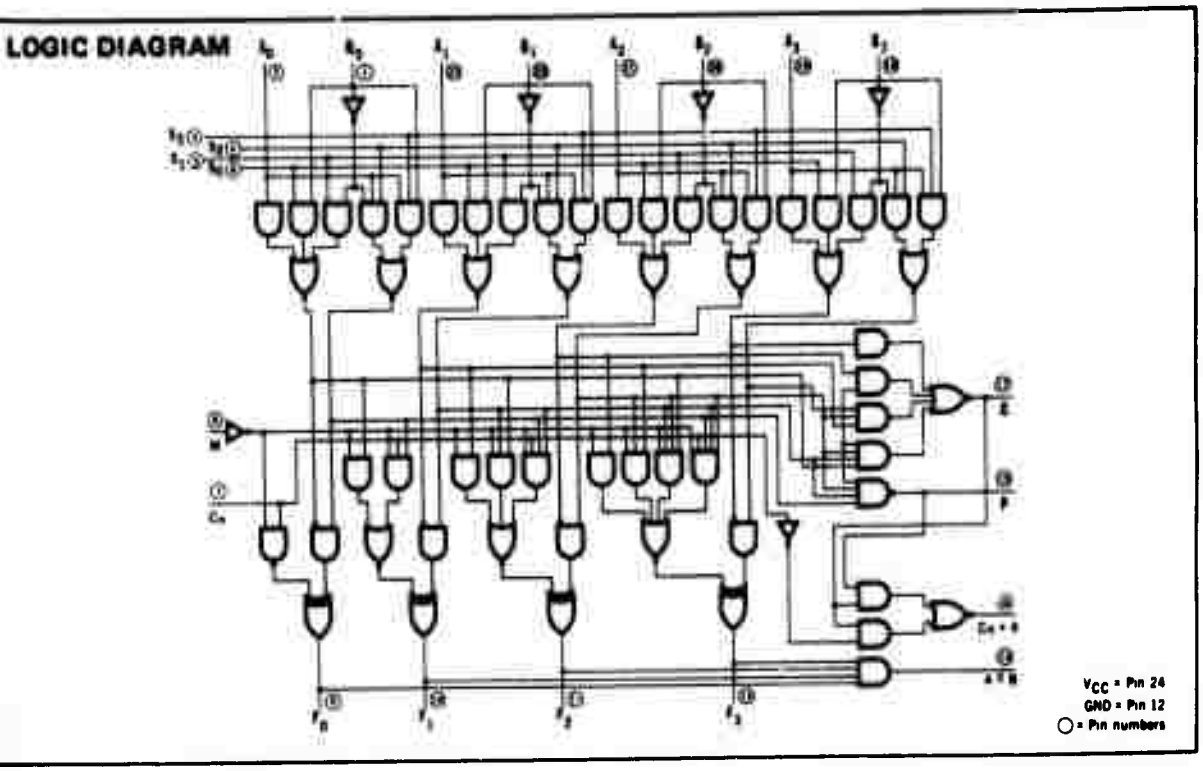

Fig. XI-5. Logic diagram for the 74181 Arithmetic Logic Unit.

BASIC FUNCTION	CONTROL SIGNALS					
	S_0	S_1	S_2	S_3	M	$\overline{C}_0$
0	1	1	0	0	1	x
A	0	0	0	0	0	1
B	0	1	0	1	1	x
A+B	1	0	0	1	0	1
A-B	0	1	1	0	0	0
CONTROL FUNCTION						
A+B	1	0	0	1	0	1
A-B	0	1	1	0	0	0
A+(B if y)	y	0	0	y	0	1
(A if z)+(B if y)	$\overline{z} \oplus y$	$\overline{z}$	0	y	$\overline{z}$	1

Fig. XI-6. Control functions for the 74181 Arithmetic Logic Unit.

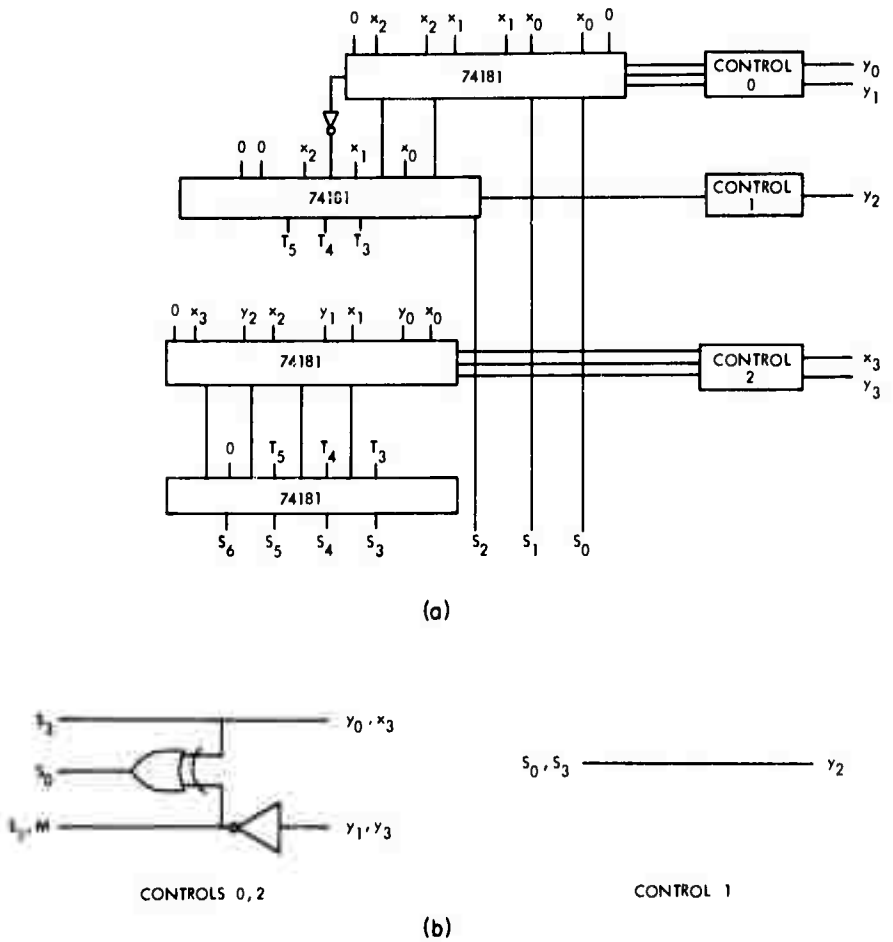

Fig. XI-7. Four by four multiplier block diagram, with external control circuitry.

in Schottky TTL. This is considerably faster than the performance obtainable from specialized multiplier packages, such as the Fairchild 9344 or the Advanced Micro Devices AM 2505. Since the ALU package has many uses, it is relatively inexpensive, particularly considering the resulting multiplier speed. The package count, and hence power, is high (approximately N(N+1)/4 for an N×N multiply; for N = 16, 69 ALU's were required) but layout is simple, and no other design incorporating standard commercial MSI packages has been able to yield the speed of this ALU array.

Certainly faster or cheaper multipliers have been built. The ALU in a binary tree, however, appears to be an optimal choice when very high speed is desired from standard commercial packages.

J. Allen, E. R. Jensen

References

1. S. D. Pezaris, "A 40-ns 17-bit by 17-bit Array Multiplier," IEEE Trans. on Computers, Vol. C-20, No. 4, pp. 442-447, April 1971.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
RESEARCH LABORATORY OF ELECTRONICS
Cambridge, Massachusetts 02139

Reprinted from
Quarterly Progress Report No. 105, April 15, 1972

XIII. SIGNAL PROCESSING*

Academic and Research Staff

Prof. A. G. Bose	Prof. A. V. Oppenheim	Prof. H. J. Zimmermann
Prof. J. D. Bruce	Prof. C. L. Searle	Dr. M. V. Cerillo

Graduate Students

J. B. Bourne	D. A. Feldman	J. R. Samson, Jr.
M. F. Davis	J. M. Kates	R. M. Stern, Jr.
	R. M. Mersereau	

A. RECENT ADVANCES IN THE THEORY OF RECONSTRUCTING MULTIDIMENSIONAL SIGNALS FROM PROJECTIONS

1. Introduction

The problem of reconstructing multidimensional signals from their projections is of interest because x-ray photographs and electron micrographs can be considered to be projections of three-dimensional objects. Thus mathematical techniques for performing such reconstructions will permit us to reconstruct visually opaque objects from their x-rays at different orientations and to determine the structure of macromolecules from electron micrographs. In a previous report[1] some techniques were discussed whereby we could perform such a reconstruction; in the present report, some other more powerful algorithms will be developed. One of these algorithms, in fact, permits the reconstruction of a broad class of multidimensional signals of any dimensionality from a single one-dimensional projection.

The idea of reconstructing functions from their projections can be applied to functions of any dimensionality; however, the most interesting problems, since they have useful applications, are the two-dimensional and the three-dimensional problems. By extension, there is a one-dimensional problem, but it is a trivial case because the projection of a one-dimensional function is the function itself. Most of the derivations in this report will be given in terms of the two-dimensional problem because it is notationally and conceptually simpler than the three-dimensional problem, but we shall also explore some of the issues that are unique to the three-dimensional case.

In both of the algorithms that are developed here it is assumed that the function which is being reconstructed is bandlimited, and if a further assumption is made they will yield exact reconstructions. If these assumptions are not appropriate for the problem at hand, there are other techniques that will yield approximate reconstructions.[1-6]

*This work was supported by the Joint Services Electronics Programs (U.S. Army, U.S. Navy, and U.S. Air Force) under Contract DAAB07-71-C-0300, by the U.S. Coast Guard (Contract DOT-CG-13446-A), and by M.I.T. Lincoln Laboratory Purchase Order CC-570.

Inasmuch as we shall deal with bandlimited functions exclusively, it is appropriate to begin with a discussion of the properties of the projections of multidimensional bandlimited functions.

2. Projections of Bandlimited Functions

The assumption of bandlimitedness is not especially harsh, for although most functions that we shall reconstruct are spacelimited and hence strictly speaking not bandlimited, they are nearly so. Furthermore, if any algorithm is to be implemented on a computer, it is necessary to reconstruct a sampled multidimensional function from sampled projections. Thus bandlimitedness is implicitly assumed to a greater or lesser degree by all digital reconstruction algorithms. In these algorithms we shall explicitly assume bandlimitedness and then utilize this assumption in the design of our algorithms, with the hope that they will yield high-quality reconstructions for nearly bandlimited functions. The last premise must be verified experimentally.

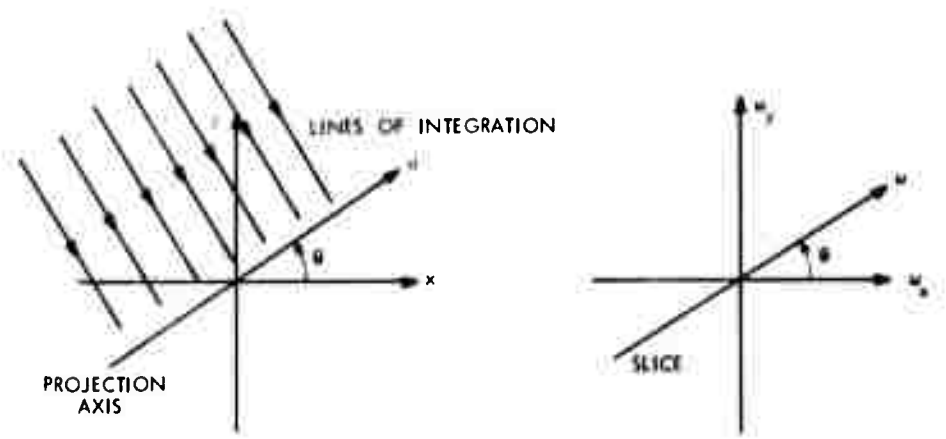

Fig. XIII-1. Relationship between a projection and a slice.

The projections of a two-dimensional function (picture) can be considered as a collection of line integrals taken perpendicular to an axis, which we call the projection axis. Thus the projection perpendicular to the x axis, $p_0(x)$, can be defined as

$$p_0(x) = \int_{-\infty}^{\infty} f(x, y)\, dy.$$

At a general angle θ, a projection can be similarly defined by

$$p_\theta(u) = \int_{-\infty}^{\infty} f(u \cdot \cos\theta + v \cdot \sin\theta, -u \cdot \sin\theta + v \cdot \cos\theta)\, dv, \qquad (1)$$

and it satisfies the Fourier transform relation

$$p_0(u) \longleftrightarrow F(\omega\cos 0, \omega\sin 0), \tag{2}$$

where $F(\omega_x, \omega_y)$ represents the two-dimensional Fourier transform of $f(x, y)$. The right-hand side of Eq. 2 will be referred to as the slice of the two-dimensional Fourier transform at an angle 0. Thus the one-dimensional Fourier transform of the projection of a picture at an angle 0 to the x axis is a slice of the two-dimensional Fourier transform of that picture at an angle 0 with the ω_x axis. This relationship is illustrated in Fig. XIII-1.

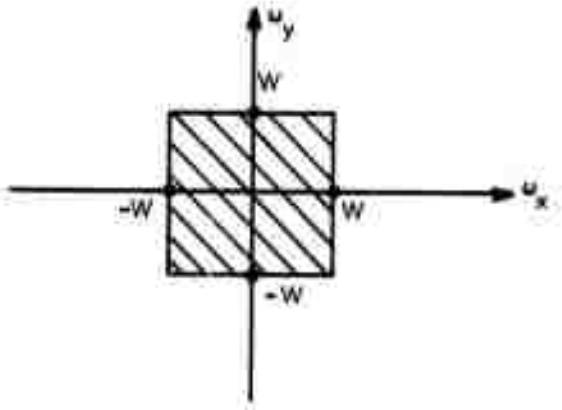

Fig. XIII-2. Region of Fourier plane over which a bandlimited picture is nonzero.

If we now assume that the picture is bandlimited, that is, that its frequency response is nonzero only in that region of the Fourier plane illustrated in Fig. XIII-2, then we can use the sampling theorem to express the picture in terms of its samples on a regular Cartesian raster as in

$$f(x, y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f\left(\frac{m\pi}{W}, \frac{n\pi}{W}\right) \frac{\sin W\left(x - \frac{m\pi}{W}\right) \sin W\left(y - \frac{n\pi}{W}\right)}{W^2\left(x - \frac{m\pi}{W}\right)\left(y - \frac{n\pi}{W}\right)}. \tag{3}$$

Since all of the projections transform to slices, they too must be bandlimited (in one dimension) and each projection can thus be expanded in terms of its samples as in

$$p_0(u) = \sum_{n=-\infty}^{\infty} p_0\left(\frac{n\pi}{W_0}\right) \frac{\sin W_0\left(u - \frac{n\pi}{W_0}\right)}{\left(u - \frac{n\pi}{W_0}\right)}. \tag{4}$$

The bandwidth of each projection W_0 can be expressed as

$$W_\theta = \frac{W}{\max\{|\cos\theta|, |\sin\theta|\}}. \tag{5}$$

From Eqs. 4 and 5 we can ascertain the Nyquist sampling rate for each projection, which is observed to be a function of θ, the projection angle. Since we must work with sampled projections, this will prove to be an important quantity.

We can get an alternative expression for $p_\theta(u)$, not in terms of the samples of the projections, but in terms of the samples of the picture itself. If we take the Fourier transform of Eq. 3, we get

$$F(\omega_x, \omega_y) = \frac{\pi^2}{W^2} \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f\left(\frac{m\pi}{W}, \frac{n\pi}{W}\right) \exp\left\{-j \frac{\pi}{W}(m\omega_x + n\omega_y)\right\} b_{WW}(\omega_x, \omega_y), \tag{6}$$

where

$$b_{WW}(\omega_x, \omega_y) = \begin{cases} 1, & \text{if } |\omega_x| \le W \text{ and } |\omega_y| \le W \\ 0, & \text{otherwise} \end{cases}$$

From Eq. 6 we can evaluate $F(\omega\cos\theta, \omega\sin\theta)$ which is the expression for a slice (from Eq. 2).

$$F(\omega\cos\theta, \omega\sin\theta) = \frac{\pi^2}{W^2} \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f\left(\frac{m\pi}{W}, \frac{n\pi}{W}\right) \exp\left\{-j \frac{\pi\omega}{W}(m\cos\theta + n\sin\theta)\right\}$$

$$\times b_{WW}(\omega\cos\theta, \omega\sin\theta). \tag{7}$$

Performing an inverse Fourier transform on Eq. 7 gives an expression for the projection at angle θ.

$$p_\theta(u) = \frac{\pi}{W^2} \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f\left(\frac{m\pi}{W}, \frac{n\pi}{W}\right) \frac{\sin W_\theta\left(u - \frac{m\pi}{W}\cos\theta - \frac{n\pi}{W}\sin\theta\right)}{u - \frac{m\pi}{W}\cos\theta - \frac{n\pi}{W}\sin\theta}. \tag{8}$$

In the two reconstruction techniques that follow, we must impose one further restriction on the picture in addition to bandlimitedness. We must assume that the digitized picture $f\left(\frac{m\pi}{W}, \frac{n\pi}{W}\right)$ be nonzero for integral values of m and n only when m and n are in the range $0 \le m, n \le N-1$, for some finite integer N. We call this assumption quasi-spacelimitedness, although note that we do not assume that f(x, y) is spacelimited (which would contradict the assumption that it is bandlimited), but only

that its samples are spacelimited. This assumption has the effect of making the double summations of Eqs. 3, 6, 7, and 8 finite. This, like the assumption of bandlimitedness, is implicit in most reconstruction techniques, since only a finite number of samples of the Fourier transform of the picture are generally computed, and only a finite number of picture samples are reconstructed.

3. An Algorithm for Reconstructing a Function from N+1 Projections

Equation 4 gives us the smallest sampling rate that can be employed for sampling a projection in order that information not be lost by sampling. Each projection, of course, can be sampled with a higher rate. The traditional approach for getting samples of the slices of a picture is to find a sampling rate that is large enough so that all of the projections can be sampled at the same rate. The resulting sequences can then be aliased to give M point sequences, and these M point sequences can then be Fourier-transformed by using a discrete Fourier transform (DFT) algorithm to yield M sample values along each slice. The M-point aliased sequence x(n) corresponding to the infinitely long sequence x(n) is defined by

$$x(n) = \sum_{m=-\infty}^{\infty} x(Mm+n).$$

If this procedure is followed, the Fourier transform of the picture will be known at points lying on a polar lattice. The points of such a lattice can be thought of as the intersections of the set of slices with a family of evenly spaced concentric circles, including one of zero radius at the origin. Once the transform of the picture is known at these points, the next step is to approximate the transform of the picture over the whole plane and then perform an inverse Fourier transform. There are no nice polar "sampling theorems" that will allow us to obtain directly the set $f\left(\frac{m\pi}{W}, \frac{n\pi}{W}\right)$.

As a different approach, let us therefore sample each projection at its own Nyquist rate, or at a rate proportional to its Nyquist rate, then alias the resulting sequences to N points (N is the width of the digitized picture) and use a DFT algorithm to get samples of the Fourier transform of the picture. If this procedure is followed, the Fourier samples which result lie at the intersection of the slices with a family of concentric squares, as illustrated in Fig. XIII-3.

In the special case of a bandlimited quasi-spacelimited (BLQSL) function, a concentric squares lattice has definite advantages over a polar one. Along any horizontal or vertical lines in the Fourier plane, the Fourier transform of a BLQSL function is a one-dimensional complex polynomial of degree N-1, and as a result any line in the Fourier plane is completely specified by N-1 samples that lie along that line. Furthermore, a BLQSL function is completely specified by its DFT, that is, by the

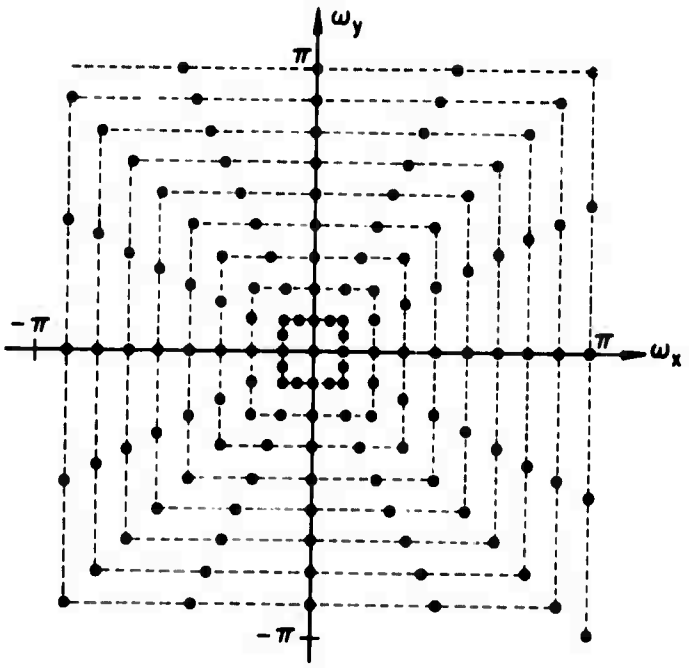

Fig. XIII-3. A set of samples of the Fourier transform of a bandlimited function obtained by sampling each projection at a rate proportional to its own Nyquist rate.

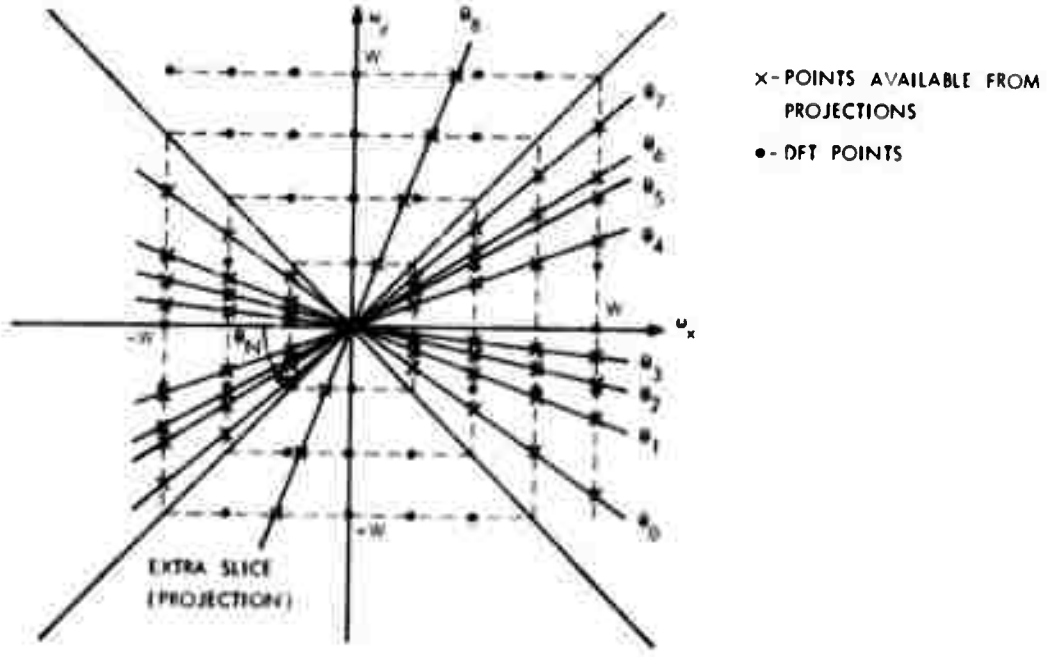

Fig. XIII-4. Set of Fourier plane samples by which an 8 × 8 picture can be reconstructed exactly, under the assumption that the picture is bandlimited and quasi-spacelimited.

samples of its Fourier transform at $F\left(\frac{W}{N} i, \frac{W}{N} j\right)$, $-\frac{N}{2} + 1 \leq i, j \leq \frac{N}{2}$. These points all lie on the sides of the concentric squares or their extensions. These properties enable us to reconstruct a BLQSL function exactly from a set of N concentric-squares projections.

Suppose, for example, we have a two-dimensional BLQSL function of dimension N. Let us assume also that we have the capability of obtaining the projections of the picture at any angle we desire. We can thus take N projections at N distinct angles in the range $-45° \leq \theta \leq 45°$, and we can also take one projection at an angle outside this range. The known points in Fourier space will then correspond to those illustrated in Fig. XIII-4 for the special case N = 8. Along each vertical square side we thus have 8 samples and along these sides the Fourier transform is a 7^{th}-order polynomial in the variable $e^{-j\omega_y}$. Then, using Lagrange polynomials (or some other technique), we can evaluate the Fourier transform at all of the DFT points on each of the vertical lines, except for the one at $\omega_x = 0$. Now consider the horizontal sides. Along each of these lines we also have a polynomial of degree 7 and we also have 8 samples, seven computed from the column calculations, and the eighth provided by the remaining projection. Since this projection was taken outside the range $-45° \leq \theta \leq 45°$, it must intersect all of the horizontal square sides (and must also not pass through any of the DFT points whose value is already known). Thus we can apply Lagrange polynomials to the horizontal lines to fill in the remaining DFT values. Consequently, we know all of the DFT values exactly, and a BLQSL picture can be

(a)

(b)

(c)

Fig. XIII-5.

Comparison of reconstructions from a concentric squares grid and from a concentric circles grid. (a) Original picture. (b) Concentric squares reconstruction. (c) Polar (concentric circles) reconstruction.

reconstructed from its DFT so that we know the set of picture samples exactly. If the last projection had been taken perpendicular to the y axis, then the second round of interpolation would not have been necessary, for the remaining DFT samples would be available directly from the DFT of the last sampled projection.

In Fig. XIII-5 we show a concentric-squares reconstruction and compare it with the corresponding concentric-circles (polar) reconstruction. Instead of using Lagrange interpolation to exactly perform the reconstruction, a simpler approximate strategy was employed. Linear interpolation was used to approximate the DFT samples from the samples obtained from the projections in both reconstructions. Each reconstruction is a 64 × 64 picture which was obtained from 64 projections. The projections were computed from the original picture which is included for comparison. Note that the concentric-squares reconstruction is truer to the original.

4. Reconstructing a BLQSL Picture from a Single Projection

Let us now restrict ourselves to the slice at an angle $\theta = \tan^{-1} 1/N$. From Eq. 7 this slice can be written as

$$F\left(\frac{N\omega}{\sqrt{N^2+1}}, \frac{\omega}{\sqrt{N^2+1}}\right) = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} f\left(\frac{m\pi}{W}, \frac{n\pi}{W}\right) \exp\left\{-j \frac{\pi\omega}{\sqrt{N^2+1}} W(Nm+n)\right\}, \quad \text{if } |\omega| \leq \frac{W}{N}\sqrt{N^2+1}$$

$$= 0 \qquad \text{otherwise} \tag{9}$$

If we define $g(Nm+n) = f\left(\frac{m\pi}{W}, \frac{n\pi}{W}\right)$, then Eq. 9 becomes

$$F\left(\frac{N\omega}{\sqrt{N^2+1}}, \frac{\omega}{\sqrt{N^2+1}}\right) = \sum_{l=0}^{N^2-1} g(l) \exp\left(-j \frac{\pi\omega l}{W\sqrt{N^2+1}}\right), \quad \text{if } |\omega| \leq \frac{W}{N}\sqrt{N^2+1}$$

$$= 0, \qquad \text{otherwise} \tag{10}$$

Thus, over the region of interest, the slice at $\theta = \tan^{-1} 1/N$ is a one-dimensional polynomial of degree N^2-1 in the variable $\exp\left(-j \frac{\pi\omega}{W\sqrt{N^2+1}}\right)$, and the coefficients of that polynomial are simply the picture samples arranged as they would be if the picture were scanned column by column. Since $F\left(\frac{N\omega}{\sqrt{N^2+1}}, \frac{\omega}{\sqrt{N^2+1}}\right)$ can be computed from

N^2 samples taken along the slice, and knowledge of a polynomial implied knowledge of its coefficients, specification of N^2 samples along the slice at $\theta = \tan^{-1} 1/N$ implies knowledge of the whole set of picture samples. (Similar statements can be made about other slices in the Fourier plane.)

Let us now, for convenience, define $G(\omega) = F\left(\frac{N\omega}{\sqrt{N^2+1}}, \frac{\omega}{\sqrt{N^2+1}}\right)$, and let us set $\Delta\omega = \frac{2W\sqrt{N^2+1}}{N^3}$. Then if we compute $G(k\Delta\omega)$ for $k = -\frac{N^2}{2} + 1, \ldots, 0, 1, \ldots, \frac{N^2}{2}$, we shall have N^2 equally spaced samples of $G(\omega)$ which extend over the entire band. There is a strong reason for choosing this particular set of frequency samples on this slice. If the projection $p_{\tan^{-1}\frac{1}{N}}(u)$ is sampled at its Nyquist rate, if the infinite sequence that results is then aliased to give a sequence of length N^2, and if this sequence is then Fourier-transformed by means of the DFT, the resulting N^2 point sequence is $G(k\Delta\omega)$. Substituting in Eq. 10, we have

$$G(k\Delta\omega) = \sum_{l=0}^{N^2-1} g(l) \exp\left(-j \frac{2\pi kl}{N^3}\right) \qquad k = -\frac{N^2}{2} + 1, \ldots, \frac{N^2}{2}. \tag{11}$$

Examining Eq. 11, we see that $G(k\Delta\omega)$ corresponds to the first N^2 points of the N^3 point DFT of the sequence formed by taking the N^2 picture samples column by column and appending $N^3 - N^2$ zeros.

The sequence $G(k\Delta\omega)$ could be obtained from the sequence $g(l)$ by means of a chirp z-transform algorithm CZT.[7] To obtain $g(l)$ (the picture samples) from $G(k\Delta\omega)$, we thus need an inverse CZT, which will be developed.

These results have an interesting interpretation in terms of another problem. The impulse response of a two-dimensional nonrecursive digital filter behaves exactly like the set of rectangular samples of a BLQSL picture and thus the impulse response, or the two-dimensional frequency response, of such a filter is completely specified by its frequency response along the line $0 = \tan^{-1} 1/N$. As well as providing an interesting property for such filters, this result suggests a mapping between one-dimensional nonrecursive and two-dimensional nonrecursive filter designs that may be useful in filter design. These implications are worthy of further study.

5. Reconstructing a Three-Dimensional BLQSL Function from a Single Projection

Probably the simplest way to reconstruct a three-dimensional sequence from its projections is to consider that three-dimensional sequence as a stack of two-dimensional

sequences. If we think of these two-dimensional sequences as lying parallel to the x-y plane, and then take a projection parallel to the x-y plane at an angle $\theta = \tan^{-1} 1/N$ with the x axis, then the resulting two-dimensional projection of the three-dimensional object will be a stack of one-dimensional projection functions, each of which is the projection of one member of the original stack of two-dimensional functions and each of which is taken at its critical angle. This is a straightforward extension of the two-dimensional problem and hardly requires elaboration. It would be a computationally efficient scheme, however, if a complete reconstruction were not desired, but only a limited number of cross sections.

From a theoretical point of view, a more interesting approach to the three-dimensional problem is to parallel the reasoning of the two-dimensional analysis. In that case we found a line in the Fourier plane: if we knew the Fourier transform of the picture along this line, then we knew the whole set of picture samples. Such a line also exists in the three-dimensional case. This is that line which is traced out by the vector $\vec{\omega}_c$, where

$$\vec{\omega}_c = \left(\frac{N^2}{\sqrt{N^4 + N^2 + 1}}, \frac{N}{\sqrt{N^4 + N^2 + 1}}, \frac{1}{\sqrt{N^4 + N^2 + 1}} \right) \omega.$$

Along this line the frequency response is a polynomial of degree $N^3 - 1$, and the coefficients of this polynomial are the function samples $f\left(\frac{m\pi}{W}, \frac{n\pi}{W}, \frac{p\pi}{W}\right)$, $0 \le m, n, p \le N-1$, where W is the bandwidth, defined as in the two-dimensional case. If we sample this line at N^3 evenly spaced points over the band, then

$$G(k\Delta\omega) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} \sum_{p=0}^{N-1} f\left(\frac{m\pi}{W}, \frac{n\pi}{W}, \frac{p\pi}{W}\right) \exp\left\{-j \frac{2\pi k}{N^5} (N^2 m + Nn + p)\right\},$$

$$k = -\frac{N^3}{2} + 1, \ldots, 0, 1, \ldots, \frac{N^3}{2}. \qquad (12)$$

Thus we have the first N^3 points of an N^5 point sequence. Equation 12 can be solved by using the inverse chirp z-transform.

The projection of a three-dimensional function is two-dimensional, whereas the critical line along which we desire the frequency response is one-dimensional. This frequency response can be evaluated directly from the two-dimensional projection samples (the projection is a bandlimited function) or equivalently a one-dimensional projection of the two-dimensional projection can be computed digitally and then transformed. If the angle of this projection is chosen properly, this slice of a slice will correspond to the desired line. It must be remembered however, when working with the

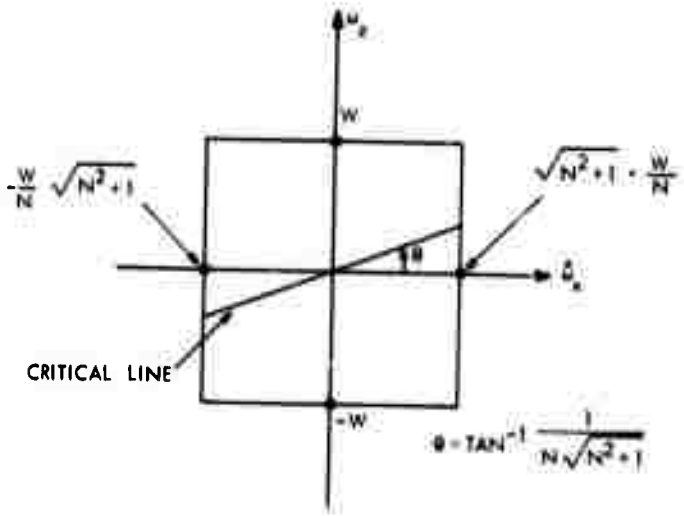

Fig. XIII-6. Two-dimensional slice of the Fourier transform of a three-dimensional function taken perpendicular to the plane $\omega_x = N\omega_y$. The bandwidths of the slice are shown, as well as the location of the critical line, whose critical frequency response determines the whole three-dimensional frequency response.

two-dimensional projection that although this is a bandlimited function, the bandwidth in the two orthogonal frequency variables is a function of the direction of that projection. In Fig. XIII-6 we show the relevant parameters for computing the frequency response along the critical line when the original projection was projected onto the plane $\omega_x = N\omega_y$.

6. Inverse Chirp z-Transform

The chirp z-transform (CZT) algorithm[7] is an efficient algorithm for evaluating the sum

$$X_k = \sum_{n=0}^{L-1} x(n)(AW^k)^n \qquad k = 0, 1, \ldots, K-1, \tag{13}$$

where

$$A = A_o \exp(j2\pi\theta_o)$$

$$W = W_o \exp(j2\pi\phi_o).$$

The CZT calculates the z-transform of the finite duration sequence x(n) at a set of points that are regularly spaced on a spiral in the z plane as illustrated in Fig. XIII-7.[7] Equation 11 can be seen to be of the same form as Eq. 13 if in place of the sequence x(n)

we substitute the sequence $g(l) = g(Nm+n) = f\left(\frac{m\pi}{W}, \frac{n\pi}{W}\right)$ and if we set $A = \exp\left(-j\frac{N^2}{2} + 1\right)$ and if $W = \exp(j2\pi/N^3)$. The sequence X_k and A and W are known in this particular case, and we desire a means of calculating g(l). What we need, therefore, is a means of inverting Eq. 13 – an inverse CZT.

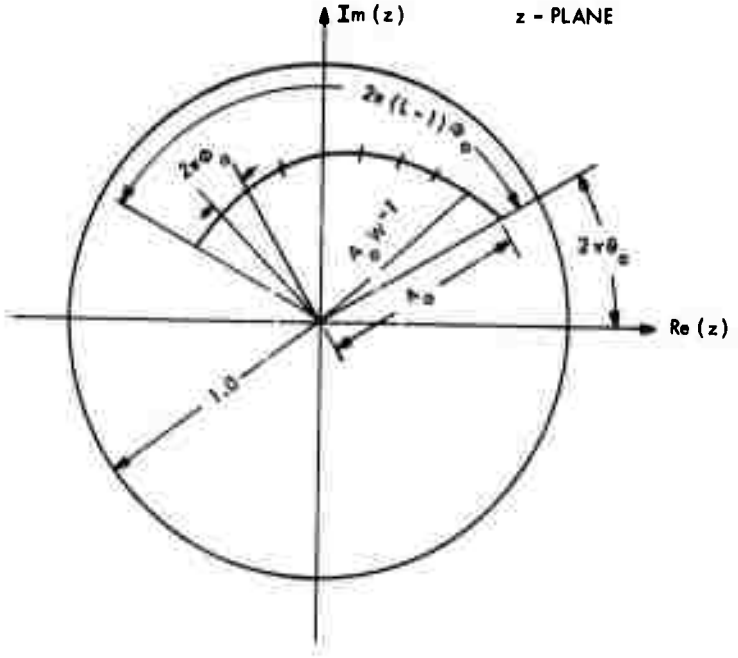

Fig. XIII-7. Illustration of the independent parameters of the CZT algorithm and the inverse CZT algorithm. (Modified from Rabiner et al.[8])

Since the sequence X_k corresponds to samples of a polynomial of degree L-1, we know that Eq. 13 can be inverted if there are more than L independent values of X_k, or if $K \geq L$. This follows from the fact that the matrix of coefficients $[(AW^k)^n]$ is a Vandermonde matrix. One possible technique to use is to invert (13) directly. For values of K of the order of several thousand, however, this is computationally not feasible.

Another approach which proves to be far more attractive computationally, although at first appearance it would not be so, is to use the Lagrange polynomial interpolation formula to reconstruct the complete polynomial over the whole z plane from the set of K samples and then perform an inverse z-transform integral of this polynomial to get the sequence x(n).

If X(z) is a polynomial of degree L-1 in z^{-1} and if X(z) is specified at the points $z_0^{-1}, z_1^{-1}, \ldots, z_{L-1}^{-1}$, then

$$X(z) = \sum_{m=0}^{L-1} X(z_m)\, \ell_m(z^{-1}), \tag{14}$$

where

$$z_m^{-1} = [AW^m]^{-1} \qquad m = 0, 1, \ldots, L-1$$

and $\ell_m(z^{-1})$ is a Lagrange interpolating polynomial

$$\ell_m(z^{-1}) = \frac{\left(z^{-1}-z_o^{-1}\right)\left(z^{-1}-z_1^{-1}\right)\cdots\left(z^{-1}-z_{m-1}^{-1}\right)\left(z^{-1}-z_{m+1}^{-1}\right)\cdots\left(z^{-1}-z_{L-1}^{-1}\right)}{\left(z_m^{-1}-z_o^{-1}\right)\left(z_m^{-1}-z_1^{-1}\right)\cdots\left(z_m^{-1}-z_{m-1}^{-1}\right)\left(z_m^{-1}-z_{m+1}^{-1}\right)\cdots\left(z_m^{-1}-z_{m+1}^{-1}\right)}.$$

Since the denominator of $\ell_m(z^{-1})$ is a constant, let us write it as $1/C_m$. Thus

$$X(z) = \left[\sum_{m=0}^{L-1} \frac{X(z_m)\, C_m}{\left(z^{-1}-z_m^{-1}\right)}\right]\left[\prod_{\ell=0}^{L-1}\left(z^{-1}-z_\ell^{-1}\right)\right]. \tag{15}$$

Equation 15 represents the z-transform of the sequence x(n) which we desire. Thus we see that the sequence x(n) can be regarded as the impulse response of a bank of resonators and a comb filter in cascade, as in Fig. XIII-8.

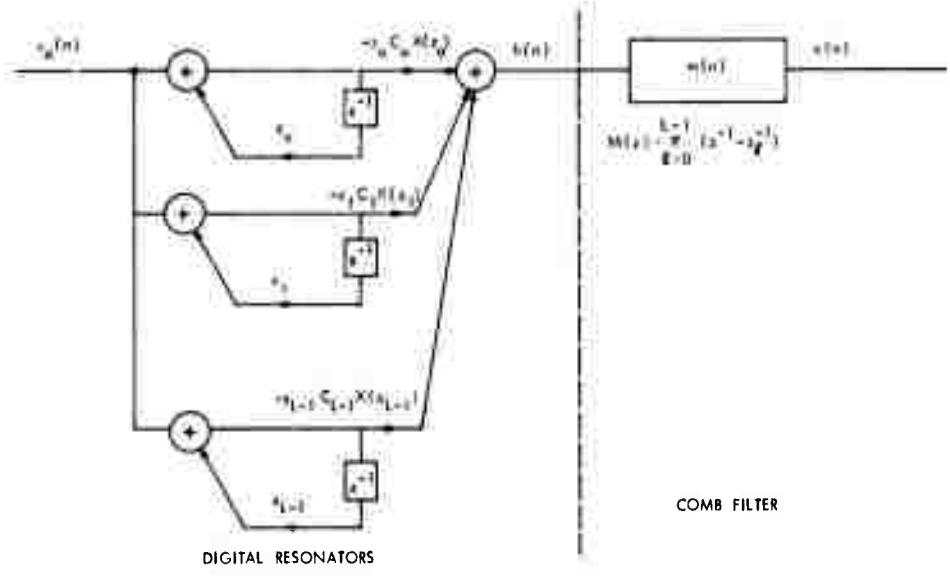

Fig. XIII-8. Digital network implementation of the inverse CZT.

Let us define h(n) to be the impulse response of the bank of resonators. From Eq. 15 we can write

$$h(n) = \sum_{i=0}^{L-1} - C_i X(z_i)\, z_i^{n+1} \qquad n = 0, 1, \ldots, L-1.$$

If we recognize that $z_i = AW^{-i}$, then

$$h(n) = \sum_{i=0}^{L-1} - C_i X(z_i)\, A^{n+1} W^{-i(n+1)} = A^{n+1} \sum_{i=0}^{L-1} - C_i X(z_i)\, W^{-i}\, W^{-in}$$

$$n = 0, 1, \ldots, L-1. \qquad (16)$$

If we write

$$CZT(x(n), A, W, N) = \sum_{n=0}^{N-1} x(n)\, A^{-n} W^{nk}, \qquad (17)$$

then we can write (16) in the form

$$h(n) = -A^{n+1} CZT(C_n X(z_n), W, W^{-1}, L) \qquad (18)$$

and thus h(n) can be evaluated efficiently by using the CZT algorithm itself.

The output sequence x(n) is then

$$x(n) = h(n) \circledast m(n),$$

where $\circledast$ denotes convolution. Inasmuch as we only care about the first L values of the sequence x(n) and m(n) is a causal sequence of length L+1, only the first L values of the sequence h(n) are necessary. This fact allows us to evaluate h(n) using a CZT, and will further allow us to perform the convolution of (18) using high-speed convolution techniques.

Except for calculating the arrays C_k and m(n), the computation of the inverse CZT can all be done efficiently. In fact, the time required to calculate an inverse CZT is approximately twice that required to calculate a CZT, and thus is roughly proportional to $2L \log_2 2L$ if L is a power of two. To the best of my knowledge, there are no particularly convenient methods for calculating C_k and m(n). These quantities do not depend upon the sequence X_k, but only on the location of the samples of the z-transform in the z plane, and therefore they will be the same for all reconstructions

of a given size, which will allow these arrays to be precomputed and stored. In this sense, the calculation of these quantities can be overlooked when talking about computation times. To reconstruct a 32 × 32 array from a single projection requires approximately 10^5 operations (complex multiplies and adds) if the calculation of these initial arrays is overlooked, and it requires approximately 5000 complex storage locations. To solve Eq. 13 by direct inversion would require approximately 10^9 operations and roughly 10^6 complex storage locations.

A one-projection reconstruction algorithm is now being implemented.

R. M. Mersereau

References

1. R. M. Mersereau, "Reconstruction of Two-Dimensional Signals from Projections," Quarterly Progress Report No. 102, Research Laboratory of Electronics, M.I.T., July 15, 1971, pp. 190-196.
2. R. A. Crowther, D. J. DeRosier, and A. Klug, "The Reconstruction of a Three-Dimensional Structure and Its Applications to Electron Microscopy," Proc. Roy. Soc. (London) A 317, 319-340 (1970).
3. D. J. DeRosier and A. Klug, "Reconstruction of Three-Dimensional Structures from Electron Micrographs," Nature 217, 130-134 (1968).
4. R. Gordon, R. Bender, and G. T. Herman, "Algebraic Reconstruction Techniques (ART) for Three-Dimensional Electron Microscopy and X-ray Photography," J. Theoret. Biol. 29, 471-481 (1970).
5. R. N. Bracewell and A. C. Riddle, "Inversion of Fan Beam Scans in Radio Astronomy," Astrophys. J., Vol. 150, No. 2, Part 1, pp. 427-434, November 1967.
6. O. Tretiak, D. Ozonoff, J. Klopping, and M. Eden, "Calculation of Internal Structure from Multiple Radiographs," Proc. Two-Dimensional Digital Signal Processing Conference, University of Missouri, Columbia, Missouri, October 7, 1971, pp. 6.2.1-6.2.3.
7. L. R. Rabiner, R. W. Schafer, and C. M. Rader, "The Chirp Z-Transform Algorithm and Its Application," Bell System Tech. J. 48, 1249-1292 (1969).
8. Ibid., Fig. 2, p. 1253. Original figure, copyright 1969, American Telephone and Telegraph Company. Printed by permission.

B. TRANSIENT RESPONSE OF A VARACTOR-CONTROLLED OSCILLATOR

A study has been made of the Q-related effects on the transient response of a voltage-controlled negative-resistance oscillator.[1] The equation governing the nonlinear oscillations of a second-order time-invariant circuit has the form

$$\frac{d^2x}{dt^2} + \omega_o^2 x = \delta f\left(x, \frac{dx}{dt}\right)$$

in which $f\left(x, \frac{dx}{dt}\right)$ is a general nonlinear function of the variable x and its derivative,

and δ is proportional to the reciprocal of the effective Q of the circuit. A perturbational analysis of this equation yields a solution in which the instantaneous frequency of oscillation ω_i is given by an expression of the form $\omega_i = \omega_o + F(a)$, where a corresponds to the magnitude of the amplitude envelope of the oscillations. This expression indicates a possible variation in frequency because of a variation in the amplitude envelope during a transient period of the oscillations. It can be argued that the frequency does not reach a steady-state value until the amplitude reaches a steady-state value.

1. Cause and Mode of Transient Operation

An analysis of an idealized step-change in one of the frequency-determining elements indicates that such a change could cause a disturbance from the equilibrium steady-state oscillation. Such a disturbance implies that the state of the oscillation, specified in the phase plane by x and dx/dt immediately after the change occurs, does not correspond in general to a state that is located on the steady-state limit cycle. Standard phase-plane analysis shows that if the state of the oscillator is described by a set of coordinates (the operating point) not located on the limit cycle, the oscillation will spiral to the stable limit cycle. This spiraling to the limit cycle corresponds to a variation in the amplitude envelope. The nonlinear mechanism that determines the steady-state limit cycle operation also controls this transient response back to the steady state.

2. Specific Case – Van der Pol Negative-Resistance Oscillator

A specific case of an oscillator with a Van der Pol type of nonlinearity was studied,

$$f\left(x, \frac{dx}{dt}\right) = (1-x^2)\frac{dx}{dt}.$$

The analytical solution to a second-order approximation indicated that the parameter $\delta\left(\propto \frac{1}{Q}\right)$ has a strong influence on the transient response, a first-order effect on the rate of amplitude variation, and a second-order effect on the frequency. These relations take the form $da/dt = \delta B(a)$, and $\omega_i = \omega_o + \delta^2 K(a)$, where the B(a) and K(a) are specific functions of a, corresponding to the solution of the Van der Pol equation.

It was noted that the lower the Q of the circuit, the faster the response back to the steady state following some disturbance, but at the expense of frequency stability and noise reduction properties of the oscillator in the steady state.

3. Lower Limit of Transient Response Time

Specific consideration of a varactor-controlled oscillator established a lower limit for the transient response time. First, consider the case of a circuit with an infinite Q, the harmonic oscillator, governed by the second-order differential equation

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
RESEARCH LABORATORY OF ELECTRONICS
Cambridge, Massachusetts 02139

Reprinted from
Quarterly Progress Report No. 104 April 15, 1972

occurs during a synthesis run is reported with the block identifier showing where it happened. If this occurs, signal levels in the vicinity of the offending block should be reduced by modifying the configuration and/or inputs to a level just under that which causes overflow. The only synthesizer processing blocks in which overflow can occur are addition-type blocks and filters.

One can easily envision achieving real-time synthesis by doing the final signal processing in hardware, either analog (for example, controlling Moog or Buchla modules) or digital (which could be designed to be much more flexible and would be inherently more stable). The cost of digital hardware continues to decrease rapidly, which suggests that a digital hardware synthesizer will soon be economically feasible. At the present time we are making a modest effort to design and construct such digital hardware. With a real-time synthesizer additional real-time (at "performance" time) control inputs would become possible. The distinction between notation inputs and real-time inputs would be somewhat analogous to the distinction between a composer and a performer or conductor.

Richard E. Albright's thesis research[5] contributed significantly to the evolution of MITSYN. Many conversations and work sessions with Robert P. Ceely, a composer who has been MITSYN's most demanding user,[6] have also stimulated the work.

W. L. Henke

References

1. M. V. Mathews, The Technology of Computer Music (The M.I.T. Press, Cambridge, Mass., 1969).

2. M. V. Mathews and F. R. Moore, "GROOVE – A Program to Compose, Store, and Edit Functions of Time," Communs. ACM 13, 715-721 (1970).

3. W. L. Henke, MITSYN User's Manual (unpublished, available from the author, Research Laboratory of Electronics, M.I.T., Cambridge, Mass. 02139).

4. W. L. Henke, "An Interactive Computer Graphic and Audio System Sans Large Budgets and Great Fuss," Quarterly Progress Report No. 98, Research Laboratory of Electronics, M.I.T., July 15, 1970, pp. 126-133.

5. R. E. Albright, "Methods of Tone Synthesis in Computer Music," S.M. Thesis, Department of Electrical Engineering, M.I.T., September 1970.

6. R. P. Ceely, "A Composer's View of MITSYN," Preprint No. 811 (M-1), Audio Engineering Society Convention, October 1971.

C. SPEECH ANALYSIS BY LINEAR PREDICTION

1. Introduction

This report describes the development of a speech analysis system based on linear prediction of the speech wave. The analysis is achieved by representing the speech wave

in terms of a set of parameters closely related to the glottal excitation function and the vocal-tract transfer function.

The system has been implemented by utilizing the computer facilities of Group 24 at Lincoln Laboratory, M.I.T. These facilities include the Univac 1219 computer, which is a medium-sized general-purpose computer; the Fast Digital Processor, which is a fast programmable signal processor attached to the Univac 1219; and peripherics, such as A/D and D/A converters and various display facilities. The system is capable of performing real-time spectrum analysis when both spectral cross-section and spectrographic displays are possible. Effort is now directed toward evaluation of its performance in extracting such acoustic parameters as formants and fundamental frequency of voicing. An initial attempt at formant tracking on spectra derived from linear prediction has given promising results.

We shall review briefly the theory of linear prediction, describe the implemented system, and give some preliminary results of speech analysis using this system.

2. Theory

Detailed treatments of the theory of linear prediction and its variations have been reported.[1-4] Our analysis is based on the speech-production model shown in Fig. IX-6. The all-pole digital filter H(z) represents the combined effect of the glottal source, the

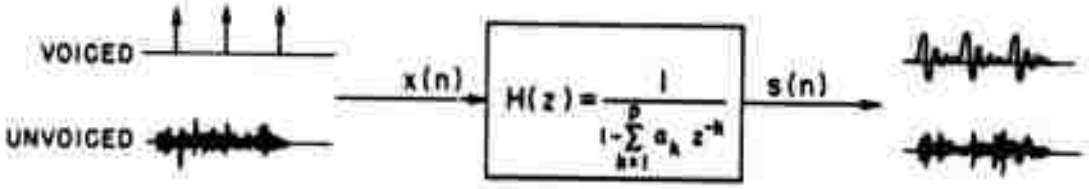

Fig. IX-6. Model of speech production.

vocal tract, and radiation losses. In this idealized model the filter is excited either by a periodic impulse train for voiced speech or random noise for unvoiced speech.

The speech production model can be equivalently characterized by the difference equation

$$s(n) = \sum_{k=1}^{p} a_k s(n-k) + x(n), \qquad (1)$$

where s(n) and x(n) are the n^{th} samples of the output speech wave and the input, respectively. The a_k's are the coefficients characterizing the filter H(z), and henceforth will be referred to as the predictive coefficients.

From Eq. 1 it is clear that we can determine the a_k's if we know the input and 2p consecutive values of s(n). The first p of these values serves as initial conditions. We shall restrict ourselves here to voiced speech in which the input is a periodic impulse train. In this case the a_k's can be determined with knowledge of only 2p consecutive values of s(n) and the position of the impulses. For this idealized model, we can define the predicted value of s(n) as

$$\hat{s}(n) = \sum_{k=1}^{p} a_k s(n-k). \tag{2}$$

The difference between s(n) and $\hat{s}(n)$ will be zero except for one sample at the beginning of each period.

In reality, however, s(n) is not produced by this highly idealized model and therefore prediction of s(n) based on Eq. 2 will introduce error. If we are to approximate s(n) by $\hat{s}(n)$ as defined by Eq. 2, the a_k's can only be determined with the specification of an error criterion.

We can choose to determine the a_k's by minimizing the mean-squared difference between s(n) and $\hat{s}(n)$, that is, by minimizing

$$E = \sum_{n=0}^{N-1} [s(n)-\hat{s}(n)]^2. \tag{3}$$

Note that the squared difference is summed over all samples except one at the beginning of each period, and we have assumed that the minimization is to be carried out over a section of s(n) of length N. It is also important to note that p more values of s(n) are needed for proper boundary conditions.

The minimum mean-squared error criterion is chosen instead of other error criteria because the determination of the a_k's now reduces to the solution of the following set of linear equations.

$$\sum_{k=1}^{p} a_k \phi_{jk} = \phi_{j0} \qquad j = 1, 2, 3, \ldots, p, \tag{4}$$

where

$$\phi_{jk} = \sum_{n=0}^{N-1} s(n-j)\, s(n-k) \qquad k = 0, 1, 2, \ldots, p. \tag{5}$$

Note that the sum in Eq. 5 excludes one point at the beginning of each period.

Equation 4 can be written in matrix form as

$$\Phi \underline{a} = \underline{\psi}, \tag{6}$$

where Φ is a $p \times p$ matrix with typical element ϕ_{jk}; $\underline{a}$ and $\underline{\psi}$ are p-dimensional vectors with the j^{th} component given by a_j and ϕ_{j0}, respectively. The solution of this matrix equation is greatly simplified by the fact that the matrix is symmetric and hence recursive procedures are applicable.

It is of interest to compare the analysis procedure outlined above for two different cases. If the fundamental frequency of voicing is known in advance, the analysis can be carried out directly, in the sense that Eq. 5 can be evaluated exactly. In practice, however, it is highly desirable to carry out the analysis without a priori knowledge of pitch. In this case an approximation has to be made and additional error is introduced. We shall illustrate this point by a simple example, but the argument can easily be generalized to include more complicated situations.

Let us assume that there is only one pitch pulse in the data and it occurs at $n = m$. If m is known, then Eq. 5 can be evaluated as

$$\phi_{jk} = \sum_{\substack{n=0 \\ n \neq m}}^{N-1} s(n-j)\, s(n-k). \tag{7}$$

Equation 5 can not be evaluated explicitly, however, if m is unknown.

Let us now approximate ϕ_{jk} by

$$\hat{\phi}_{jk} = \sum_{n=0}^{N-1} s(n-j)\, s(n-k). \tag{8}$$

Comparing Eqs. 7 and 8, we find that the error in ϕ_{jk} is given by

$$\epsilon_{jk} = \hat{\phi}_{jk} - \phi_{jk} = s(m-j)\, s(m-k). \tag{9}$$

By the nature of the speech wave, $s(m-j)$ and $s(m-k)$ are small compared with samples at the beginning of each period. Therefore the error ϵ_{jk} is small compared with ϕ_{jk} for any reasonable N. Results of comparing the two analysis procedures will be presented.

The theory of linear prediction has also been formulated in a slightly different way.[3,4] Let $e(n)$ denote the output of the inverse filter $H^{-1}(z)$ when it is excited by $s(n)$. If we choose to determine the a_k's by minimizing the total energy in $e(n)$, the set of equations obtained can be shown to be almost identical to Eqs. 4 and 5. The only difference between the two formulations is that, since $e(n)$ is of length $N + p$, the matrix Φ in the second formulation is of Toeplitz form,

$$\phi_{jk} = \phi_{|j-k|,\,0} \tag{10}$$

Although the theory developed thus far is for voiced speech, we have used the same procedure to determine the predictive coefficients for unvoiced speech.

3. Speech System

Figure IX-7 is a block diagram of the analysis system. At present, only that part of the system enclosed in the dashed lines has been implemented. Input data are first pre-emphasized (10 dB/octave), bandlimited to 5 kHz, and sampled at 10 kHz. The computation of the $\hat{\phi}_{jk}$, as defined by Eq. 8, can be greatly reduced by noting that

$$\hat{\phi}_{j+1,\,k+1} = \hat{\phi}_{jk} + s(-1-i)\, s(-1-j) - s(N-1-i)\, s(N-1-j). \tag{11}$$

Therefore only $\hat{\phi}_{j0}$ for $j = 0, 1, 2, \ldots, p$ need be computed directly. These are the first p+1 points of the short-time autocorrelation function of s(n). The rest of the matrix elements are obtained recursively from Eq. 11. The last two terms on the right-hand side of Eq. 11 can vanish to result in a Toeplitz matrix, depending on how the problem is formulated. After the elements of the matrix are formed, Eq. 6 is solved by the method of square-rooting.[5]

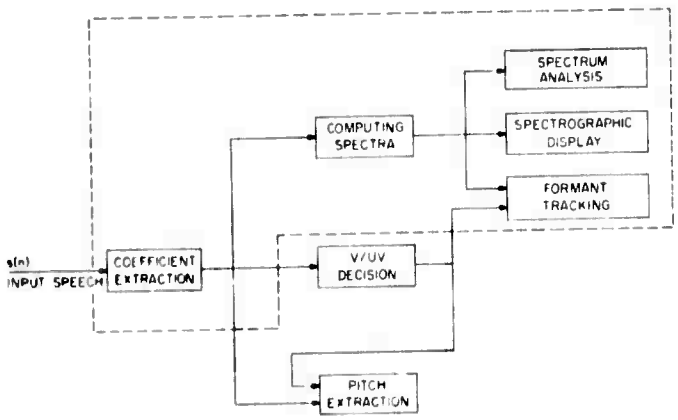

Fig. IX-7. Analysis system.

From the predictive coefficients, the approximated spectral envelope of s(n) can then be computed as $|H(e^{j\omega})|$. Note that the unit-sample response of the inverse filter $H^{-1}(z)$ is given by

$$\hat{h}(n) = \begin{cases} 1 & \text{for } n = 0 \\ a_n & \text{for } n = 1, 2, 3, \ldots, p \\ 0 & \text{otherwise} \end{cases}$$

Therefore $|H(e^{j\omega})|$ can be obtained efficiently by computing the discrete Fourier transform of $\hat{h}(n)$ with a fast Fourier transform algorithm, and then inverting the result. Each spectral cross section is multiplied by the rms value of the input data to provide gain normalization.

Both the input data length N and the order of the filter p are variables; the choice of these variables has been discussed elsewhere.[1,3] Unless otherwise specified, all results presented are obtained with n = 256 and p = 12. The coefficients are recomputed every 6.4 ms.

4. Preliminary Results

In Fig. IX-8 spectra of a synthetic vowel /a/ obtained by using various techniques are compared: (a) and (b) by windowing (with different window widths) and Fourier-transforming the waveform, (c) by cepstral smoothing,[6] and (d) by linear prediction. In Fig. IX-8a the effect of glottal periodicities can be seen as the ripples superimposed on the spectral envelope. These ripples are greatly reduced in Fig. IX-8b because of spectral smearing of the wider frequency window. In Fig. IX-8c the effect of glottal periodicities is removed by a homomorphic technique. This effect is also removed in Fig. IX-8d. But, since the analysis is based on a specific model and thus limits the number of spectral peaks, there are no extraneous peaks in Fig. IX-8d. If we compare the locations of the spectral peaks with the actual values of the five formants, it is clear that, for this example, the spectrum derived from linear prediction provides accurate formant locations.

Figure IX-9 shows the spectrum of the same vowel obtained by linear prediction, except that in this case the analysis is carried out pitch-synchronously. Comparing Figs. IX-8d and IX-9, except for the bandwidth of the second spectral peak, we find that the qualitative difference between the two spectra is quite small.

It should be noted that we have chosen to use a lot of synthetic speech material in our study. This is because parameters of synthetic speech are known exactly. Therefore the use of synthetic speech can provide us with a more objective evaluation of the analysis system.

Figure IX-10 is a spectrographic display of a sentence generated from a synthesis-by-rule program developed by D. H. Klatt. Some observations can be made concerinig Fig. IX-10. First of all, the smooth and continuous formant trajectories are clearly visible for all non-nasal sonorants. Second, the analysis is able to separate closely spaced formants very well, as in the case of /r/. The analysis also worked well for fricatives, nasals, and stops, in the sense that spectra obtained during frication, nasalization, and aspiration contain the important features characterizing these phonemes. For example, spectra derived from linear prediction for nasals all have a low-frequency peak, followed by a relative absence of energy in the 500 ~ 1500 Hz

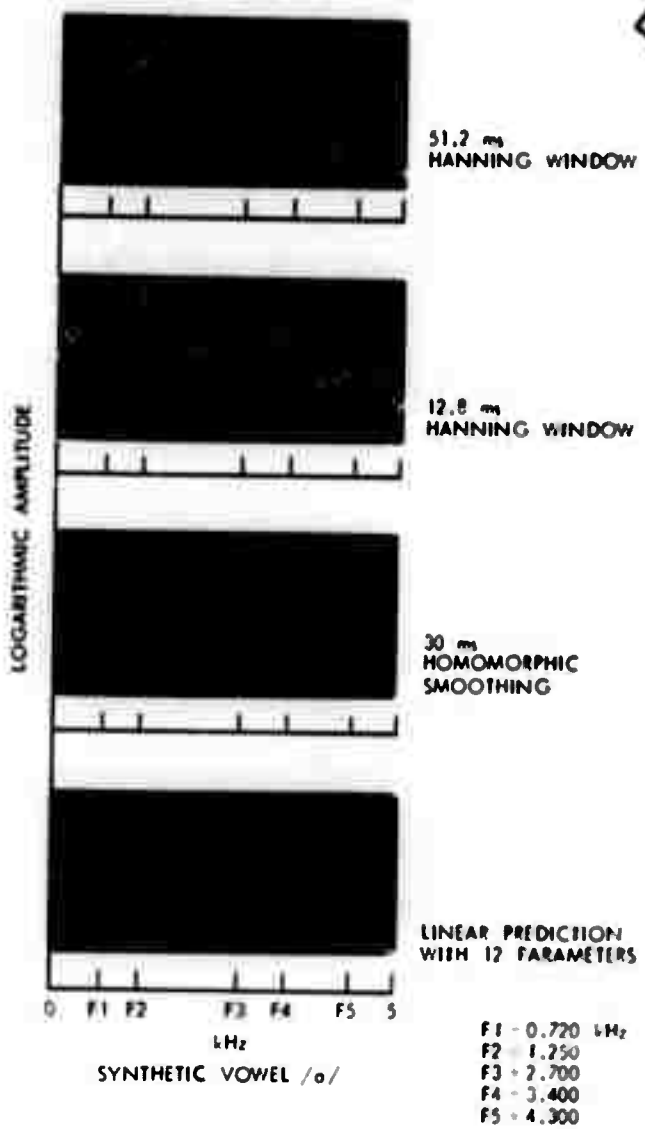

Fig. IX-8.

Spectra of synthetic /a/: (a) by windowing (51.2 ms) and Fourier transforming. (b) by windowing (12.8 ms) and Fourier transforming. (c) by cepstral smoothing. (d) by linear prediction (N = 128).

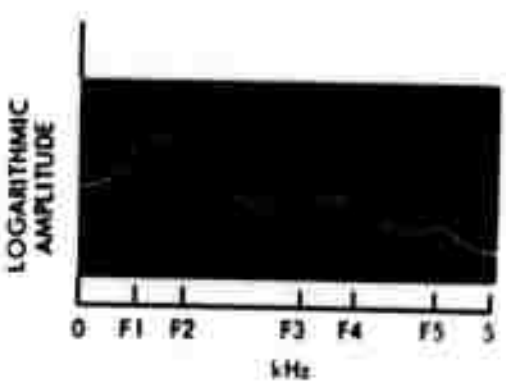

Fig. IX-9.

Spectrum of synthetic /a/ from pitch-synchronous linear prediction analysis (N = 128).

region. These are some of the important spectral attributes of the nasals.

Figure IX-11 is a spectrographic display of a sentence spoken by a male subject and has features similar to those discussed above.

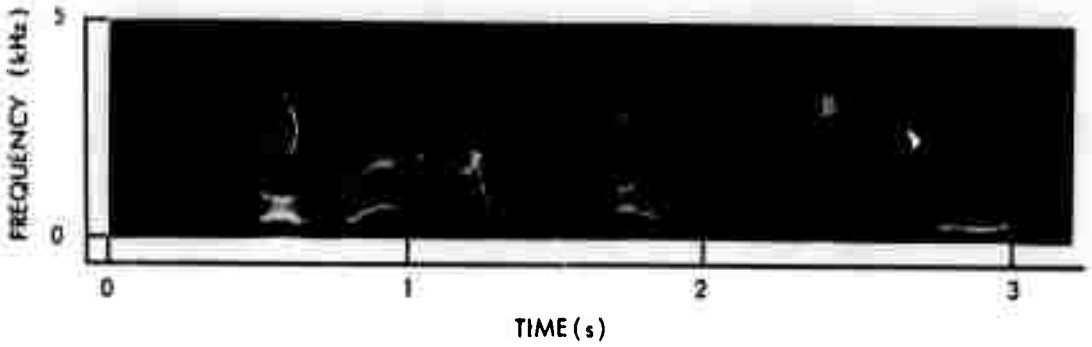

Fig. IX-10. Spectrographic display of the sentence "This program synthesizes speech by rule." (Synthetic speech.)

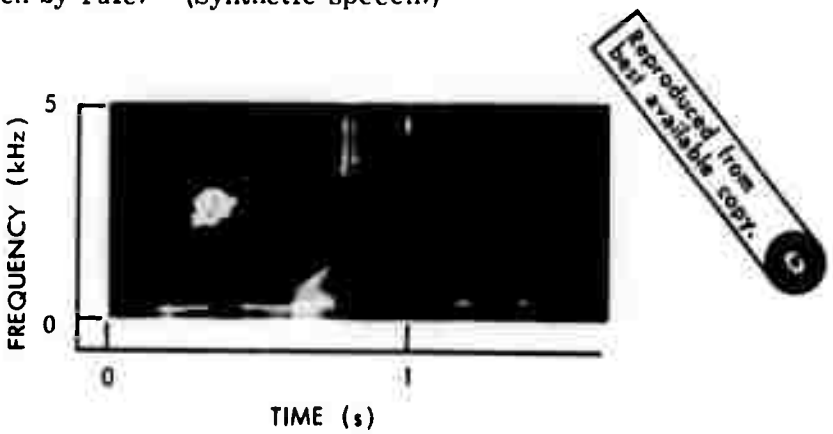

Fig. IX-11. Spectrographic display of the sentence "Can you be more specific?" spoken by a male subject.

From Figs. IX-10 and IX-11 it is clear that during the voiced portion of speech the formants are sharply defined and their trajectories are smooth and continuous. It is therefore reasonable to expect formant tracking by a simple peak-picking algorithm to give good results. Although results of this are not included in this report because a voiced-unvoiced decision has not yet been implemented, formant tracking by a simple peak-picking algorithm worked well in a few examples that were tried.

The system provides highly interactive analysis and display and is capable of real-time processing. Figure IX-12 is another example to illustrate the highly interactive display capabilities of this system. The sentence is spoken by a male subject. By setting the pointer to a specific instant of time on the spectrographic display, we can display and examine the next twelve cross sections on the other oscilloscope.

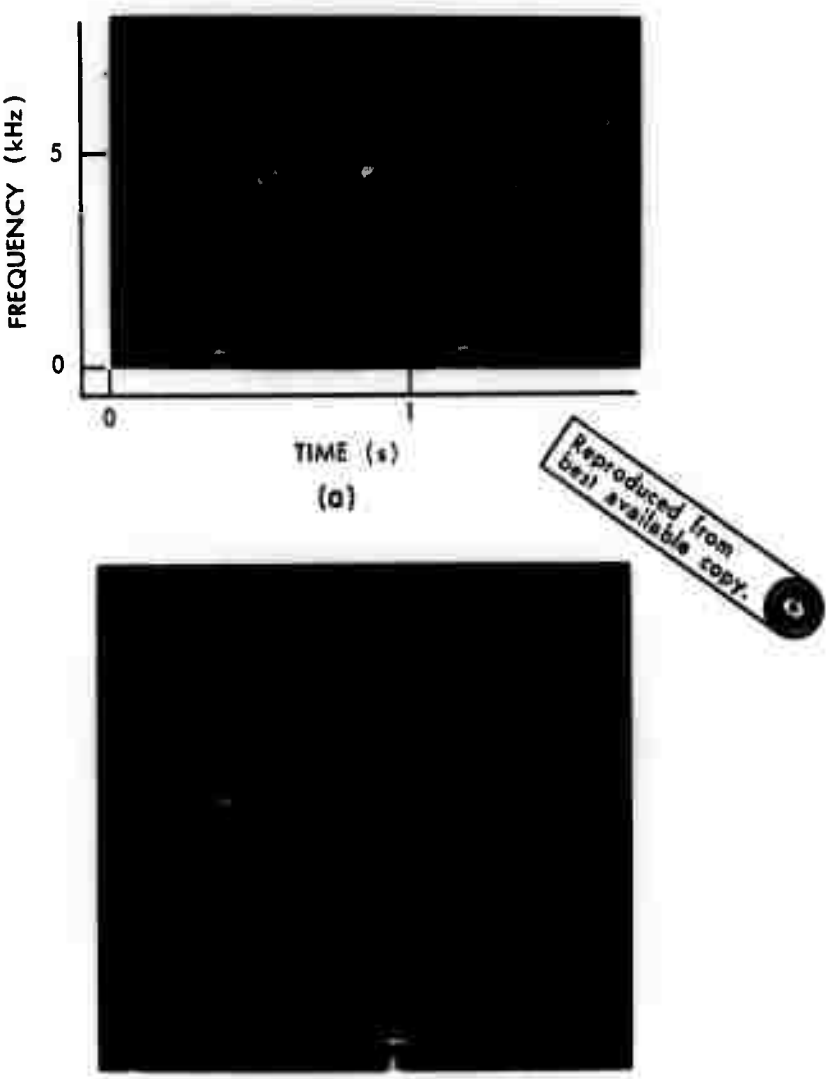

Fig. IX-12. (a) Spectrographic display of the phrase "Digital signal processing" spoken by a male subject; (b) 12 cross sections starting from the pointer in (a).

5. Summary

We have partially implemented a speech-analysis system based on linear prediction of the speech wave. The analysis technique differs from all other techniques, in that it is closely tied to a speech-production model. Our limited experience with the system indicates that it is well suited to spectrum analysis and is potentially very useful for formant tracking. The voiced-unvoiced decision and fundamental-frequency extraction parts of the system are now being implemented.

The fact that the analysis is based on a specific speech production model also imposes limitations on the technique. It is well known that during the production of nasals and fricatives there exist zeros as well as poles in the vocal-tract transfer function. It can be argued that we can always approximate these zeros by multiple poles and that the important features characterizing these phonemes are generally contained in the overall shape, not in the specific pole-zero locations, of the spectrum. There are other unsettled issues, such as whether the input speech should be windowed, which of the two formulations should be chosen for actual implementation, and so forth. We are now evaluating the system with synthetic-speech material, with all parameters such as formants and bandwidths known exactly. We believe that this evaluation, together with speech synthesis based on linear prediction, will help us resolve some of these issues.

We hope that this system can serve as the acoustic parameter extraction stage of a speech-recognition system. Although it is premature to speculate on its performance for acoustic parameter extraction, the highly interactive analysis and display facilities now developed have proved to be useful in studying the spectral characteristics of phonemes and the spectral changes from coarticulations.

Programming consultation with Mrs. Stephanie McCandless is gratefully acknowledged.

V. W. Zue

References

1. B. S. Atal and Suzanne L. Hanauer, "Speech Analysis and Synthesis by Linear Prediction of the Speech Wave," J. Acoust. Soc. Am. 50, 637-665 (1971).

2. C. J. Weinstein and A. V. Oppenheim, "Predictive Coding in a Homomorphic Vocoder," IEEE Trans. on Audio and Electroacoustics, Vol. AU-19, pp. 243-248, 1971.

3. J. Markel, "Formant Trajectory Estimation from a Linear Least-Squares Inverse Filter Formulation," Monograph No. 7, Speech Communications Research Laboratory, Inc., Santa Barbara, California, 1971.

4. John Makhoul, "Aspects of Linear Prediction in the Spectral Analysis of Speech," paper to be presented at IEEE-AFCRL 1972 International Conference on Speech Communication and Processing, Boston, Mass., April 24-26, 1972.

5. V. N. Faddeeva, "Computational Methods of Linear Algebra," English Translation by C. D. Benster (Dover Publications, Inc., New York, 1959), pp. 81-85.

6. A. V. Oppenheim, "Speech Analysis-Synthesis System Based on Homomorphic Filtering," J. Acoust. Soc. Am. 45, 459-462 (1969).

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
RESEARCH LABORATORY OF ELECTRONICS
Cambridge, Massachusetts 02139

Reprinted from

Quarterly Progress Report No. 105, June 15, 1972

References

1. E. C. Cherry and W. K. Taylor, "Some Further Experiments upon the Recognition of Speech, with One and with Two Ears," J. Acoust. Soc. Am. 26, 554-559 (1954).
2. A. W. F. Huggins, "Distortion of the Temporal Pattern of Speech: Interruption and Alternation," J. Acoust. Soc. Am. 36, 1055-1064 (1964).
3. A. W. F. Huggins, "The Perception of Temporally Segmented Speech," Quarterly Progress Report No. 103, Research Laboratory of Electronics, M.I.T., October 15, 1971, pp. 126-129; Proc. VIIth International Congress of Phonetic Sciences (Mouton, The Hague, in press).

B. SOME CONSIDERATIONS IN THE USE OF LINEAR PREDICTION FOR SPEECH ANALYSIS

1. Introduction

Recently, the analysis of speech by means of a technique referred to as linear prediction, predictive coding, or least-squares inverse filtering has received considerable attention.[1-5] This technique is directed toward modeling a sequence as the output of an all-pole digital filter. When the sequence to be modeled is specified over the domain of all integers n, there is a well-defined formulation of the technique. When only a segment of the sequence is available, however, which is always the case in practice, there are several formulations of the technique that are closely related but have important differences. One objective of this report is to summarize these differences and their implications.

When the sequence of data to be modeled is of finite length and, over the interval for which it is specified, corresponds exactly to the unit-sample response of an all-pole filter, the parameters of the model obtained by using linear prediction may be nonunique. If the data correspond closely, but not exactly, to the unit-sample response of an all-pole filter, then the solution will be unique, but the unit-sample response of the resulting filter may be considerably different from the data and small changes in the data will result in large changes in the parameters of the model and its unit-sample response. A second objective of this report is to discuss this property of the technique.

2. Formulation of the Linear-Prediction Problem

We shall consider the formulation of the technique for two problems. In problem A the data are specified for all n, and in problem B only a finite segment of the data is available.

a. Problem A

Consider a sequence $s(n)$ defined for all n and for which $s(n) = 0$ for $n < 0$. We seek

an all-pole filter with transfer function $\hat{S}(z) = \dfrac{a_o}{1 - \sum_{k=1}^{p} a_k z^{-k}}$ such that its unit-sample response $\hat{s}(n)$ approximates $s(n)$. From the form of $\hat{S}(z)$, $\hat{s}(n)$ for $n > 0$ is given by

$$\hat{s}(n) = \sum_{k=1}^{p} a_k \hat{s}(n-k). \tag{1}$$

In the linear-prediction technique we define a predicted value of $s(n)$, denoted by $\tilde{s}(n)$, as

$$\tilde{s}(n) = \sum_{k=1}^{p} a_k s(n-k) \tag{2}$$

and choose the parameters a_k to minimize the error $\mathcal{E}$ defined as

$$\begin{aligned} \mathcal{E} &= \sum_{n=1}^{\infty} |s(n)-\tilde{s}(n)|^2 \\ &= \sum_{n=1}^{\infty} \left| s(n) - \sum_{k=1}^{p} a_k s(n-k) \right|^2 . \end{aligned} \tag{3}$$

We note that the sum on n excludes the origin because $s(o)$ depends only on a_o and cannot affect the result of the minimization.

By setting $\partial\mathcal{E}/\partial a_j$ to zero for $j = 1, 2, \ldots, p$, we arrive at the following set of linear equations:

$$\sum_{k=1}^{p} a_k \sum_{n=1}^{\infty} s(n-j)\, s(n-k) = \sum_{n=1}^{\infty} s(n)\, s(n-j) \qquad j = 1, 2, \ldots, p. \tag{4}$$

In matrix notation,

$$\underline{\Phi}\,\underline{a} = \underline{\psi}, \tag{5}$$

where

$$\phi_{ij} = \sum_{n=1}^{\infty} s(n-i)\, s(n-j). \tag{6}$$

$$\psi_j = \phi_{oj}.$$

It can be shown that the matrix is symmetric and Toeplitz; that is,

$$\phi_{ij} = \phi_{ji}$$

and

$$\phi_{i+1,j+1} = \phi_{ij}.$$

Therefore the solution of Eq. 5 is computationally straightforward and efficient.[6]

We shall examine some of the properties of this solution.

i. If s(n) is indeed the unit-sample response of an all-pole filter $s(n) = \sum_{k=1}^{q} b_k s(n-k)$, then with $q \leq p$ the procedure leads to the unique solution $a_k = b_k$ for $k = 1, 2, \ldots, q$, and $a_k = 0$ for $k > q$.

ii. It can be shown that the solution $\hat{S}(z)$ corresponds to a stable filter.[7]

iii. It can be shown that the error $\mathcal{E}$ is monotonic with p.

iv. Let us define the autocorrelation function of $\hat{s}(n)$ as

$$\hat{R}_j = \sum_{n=0}^{\infty} \hat{s}(n)\, \hat{s}(n-j) \tag{7}$$

and the autocorrelation function of s(n) as

$$R_j = \sum_{n=0}^{\infty} s(n)\, s(n-j). \tag{8}$$

It can be shown that $\hat{R}_j$ and R_j are related by $\hat{R}_j = [\hat{R}_o/R_o]R_j$ for $j = 1, 2, \ldots, p$.

v. Minimizing Eq. 3 is equivalent to minimizing

$$\mathcal{E} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| \frac{S(e^{j\omega})}{\hat{S}(e^{j\omega})} - 1 \right|^2 d\omega, \tag{9}$$

where

$$\hat{S}(z) = \frac{a_o}{1 - \sum_{k=1}^{p} a_k z^{-k}}.$$

Therefore, the error criterion can be interpreted in a slightly different manner. Let u(n) denote the output of the filter $S^{-1}(\mathcal{Z})$ when it is excited by s(n). The linear-prediction technique then corresponds to determining the $\{a_k\}$ such that u(n) is best approximated by a unit sample at n = 0.

From Eq. 9, we see that the error is dependent on the ratio of $S(e^{j\omega})$ vs $\hat{S}(e^{j\omega})$. It is also clear that the minimization depends on both the magnitude and phase of $S(e^{j\omega})$. Since $\hat{S}(e^{j\omega})$ can be shown to have minimum phase (that is, all poles and zeros are inside the unit circle), this procedure will work best when $S(e^{j\omega})$ is also minimum-phase. Heuristically, we can argue this in the following way: Since $S(\mathcal{Z})$ is stable, we shall concern ourselves only with the zeros of $S(\mathcal{Z})$. If $S(\mathcal{Z})$ has zeros inside the unit circle, each of these zeros (excluding those at the origin) can be approximated by multiple poles by Taylor series expansion, and the approximation will improve as we increase p, the order of $\hat{S}(\mathcal{Z})$. This suggests that if s(n) is minimum-phase, the error asymptotically goes to zero as p increases. This is no longer true, however, if $S(\mathcal{Z})$ is not minimum-phase. Consequently it would be expected that if $S(\mathcal{Z})$ is not minimum-phase, the error will not asymptotically approach zero as p increases.

b. Problem B

In this case we consider a finite segment of data of length N which we wish to model as the output of an all-pole filter. Typically, this problem has been formulated in two ways.

Formulation I

The data are multiplied by a window w(n) and the N data points are numbered from n = 0 to n = (N-1). The window is of duration N so that multiplication of the data by the window results in a sequence s'(n) which is zero for n < 0 and n > (N-1). The sequence s(n) in problem A is then taken as the sequence s'(n). In this case most of the results of problem A remain unchanged, although the sum over n is now finite. The matrix is again Toeplitz and the set of equations can be solved efficiently.

This formulation is sometimes referred to as the autocorrelation method, since the matrix $\underline{\Phi}$ is an autocorrelation matrix of s(n), as in problem A. Empirically, it has been found that for small N it is necessary to multiply s(n) by a smooth window rather than simply to truncate it, in order to minimize the end effects.[3,4]

Formulation II

No assumption is made about the data outside the interval on which they are given. Specifically, the first p values of the data are taken as initial conditions and it is assumed that with n = 0 denoting the beginning of the interval on which the data are given, the input to the all-pole filter is zero for $p \leq n < N$. The error $\mathcal{E}$ is defined as

$$\mathcal{E} = \sum_{n=p}^{N-1} \left| s(n) - \sum_{k=1}^{p} a_k s(n-k) \right|^2 ,$$

where s(n) denotes the data. To minimize the error, we set $\frac{\partial \mathcal{E}}{\partial a_j} = 0$ for $j = 1, 2, \ldots, p$ and arrive at the set of equations

$$\sum_{k=1}^{p} a_k \sum_{n=p}^{N-1} s(n-k)\, s(n-j) = \sum_{n=p}^{N-1} s(n)\, s(n-j) \qquad j = 1, 2, \ldots, p$$

or, in matrix notation, $\underline{\Phi}\,\underline{a} = \underline{\psi}$, where

$$\phi_{ij} = \sum_{n=p}^{N-1} s(n-i)\, s(n-j) \qquad i = 0, 1, \ldots, p$$

and

$$\psi_j = \phi_{oj} \qquad j = 1, 2, \ldots, p.$$

This formulation is sometimes referred to as the covariance method. The resulting matrix $\underline{\Phi}$ is still symmetric but no longer Toeplitz. In fact,

$$\phi_{i+1,j+1} = \sum_{n=p}^{N-1} s(n-i-1)\, s(n-j-1) = \sum_{n=p-1}^{N-2} s(n-i)\, s(n-j)$$

$$= \phi_{ij} - s(N-1-i)\, s(N-1-j) + s(p-1-i)\, s(p-1-j). \qquad (10)$$

The last terms in Eq. 10 can be considered an end-effect correction.

Both formulations have been used by researchers, hence it is appropriate to compare their efficiency. This is shown in Table X-1. Formulation I has the following advantages. Theoretically, the stability of the resulting filter $\hat{S}(z)$ is guaranteed, (although this is not true for implementation with finite word-length computation). Increasing p from p_o to $p_o + 1$ involves only one additional iteration; therefore, it is easy to set an error threshold to select the appropriate value of p. On the other hand, Formulation II has the advantages that scaling is relatively simple for fixed-point implementation, and the computation can be carried out in-place. It has also been pointed out that the square-root method of solving the resulting set of linear equations is numerically very stable.[9]

Table X-1. Computational efficiency of Formulations I and II.

	Formulation I	Formulation II
Matrix	Toeplitz (Can Be Solved by Levinson's Method[6])	Symmetric (Can Be Solved by Square-Root Method[8])
Storage (data)	N	N
matrix equation	$4p+4$	$(p^2+3p)/2$
window	$2N$	0
Computation		
multiplies (windowing)	N	0
(compute ϕ_{ij})	$pN-p^2$	$pN+p$
(solve matrix equation)	$2p^2+5p-6$	$(p^3+9p^2+2p)/6$
divides (or inverse)	$2p$	p
square-roots	0	p

3. Application to Sequences Closely Approximating the Response of an All-Pole Filter

The linear-prediction method is most suitable for sequences that can be closely approximated as the response of an all-pole filter. Typically, the linear-prediction technique is used to determine the parameters of the all-pole filter, and spectral analysis or resynthesis is carried out by using these parameters to generate an approximation, $\hat{s}(n)$, to the data.

For problem A we assumed, by virtue of Eq. 1 and the fact that the data are zero for $n < 0$, that the input was a unit sample at $n = 0$. Thus, to generate $\hat{s}(n)$ from the parameters, we excite the all-pole filter with a unit sample. For problem B, the autocorrelation method outlined in Formulation I suggests the same procedure, since the product of the data and the window is treated as in problem A. For Formulation II in problem B, we made the assumption that for $p \leq n < N$ the filter input is zero. It is useful to consider the result of applying linear prediction to data that do correspond exactly to the output of an all-pole filter so that the data $s(n)$ satisfy the relationship

$$s(n) = \sum_{k=1}^{q} b_k s(n-k) \tag{11}$$

on a specified interval, and we choose to estimate $s(n)$ by

$$\tilde{s}(n) = \sum_{k=1}^{p} a_k s(n-k) \tag{12}$$

on that interval. For problem A the interval is $1 \leq n \leq \infty$ and if $q \leq p$, $\hat{s}(n) = s(n)$, and the coefficients a_k will be equal to b_k for $1 \leq k \leq q$ and zero for $k > q$. For problem B, the autocorrelation formulation (Formulation I) will not in general give $\hat{s}(n) = s(n)$, and the a_k will not equal the b_k because the infinite duration sequence corresponding to the data multiplied by the window no longer satisfies the relationship of Eq. 11. With $q \leq p$ the covariance method will always give $\hat{s}(n) = s(n)$ over the interval. When $p = q$ the a_k are uniquely determined and specify a system whose unit-sample response is $\hat{s}(n) = s(n)$. For $p > q$, if the data satisfy Eq. 11 for $p \leq n < N$ but not for $0 \leq n < p$, then we conjecture that the a_k are uniquely determined. Moreover, the unit-sample response of the all-pole filter, $\hat{s}(n)$, will equal $s(n)$ only if $s(n)$ corresponds to the unit-sample response of an all-pole filter. The fact that the specified data $s(n)$ satisfy the relationship of Eq. 11 does not require it to be the unit-sample response of an all-pole filter but only that it be the response to an input which is zero for $p \leq n < N$. Now let us consider the case for which the data satisfy Eq. 11 for $0 \leq n < N$; that is, all of the specified data including the initial conditions satisfy Eq. 11. With $q \leq p$ the covariance method will always give $\hat{s}(n) = s(n)$ over the interval. When $p = q$ the a_k are uniquely determined and specify a system whose unit-sample response is $\hat{s}(n) = s(n)$. When $p > q$, however, the $(p \times p)$ matrix $\underline{\Phi}$ is of rank q, which gives a $p-q$ parameter family of solutions for the a_k. For each solution vector $\underline{a}$ in the family of solutions, $\hat{s}(n) = s(n)$, but one and only one of these solutions specifies a system whose unit-sample response is $\hat{s}(n) = s(n)$. This solution is, of course, the one for which all of the a_k vanish when $k > q$.

Consider the following simple example. Let $s(n)$ be exactly the unit-sample response of the filter

$$S(z) = \frac{1}{1 - a z^{-1}}.$$

Thus $s(n) = a^n u_{-1}(n)$. Suppose $s(n)$ is estimated by the second-order linear predictor

$$\tilde{s}(n) = a_1 s(n-1) + a_2 s(n-2).$$

We choose to minimize the mean-square error on the interval $[n_0, n_0+1]$, using $s(n_0-1)$ and $s(n_0-2)$ as starting values (Formulation II). If $n_0 > 1$, then the equation $\underline{\Phi}\,\underline{a} = \underline{\psi}$ is

$$\begin{bmatrix} a^2 & a \\ a & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} a^3 \\ a^2 \end{bmatrix}.$$

Clearly, the matrix $\underline{\Phi}$ is singular and the general solution for $\underline{a}$ can be expressed as any particular solution $\underline{a}^o$ added to any linear combination of vectors spanning the null space of $\underline{\Phi}$. We choose $\underline{a}^o = \begin{bmatrix} a \\ 0 \end{bmatrix}$ which is the particular solution for which $\hat{s}(n)$ is the unit-sample response of the filter

$$H_o(z) = \frac{1}{1 - a z^{-1}}.$$

The general solution is given by

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} a \\ 0 \end{bmatrix} + c \begin{bmatrix} -a \\ a^2 \end{bmatrix},$$

where c is an arbitrary constant. The solution $\underline{a}$ thus lies along the line $a_2 = -a a_1 + a^2$ in the $a_1 a_2$ plane. To illustrate a different solution in the solution space, suppose $c = 1$. Then $a_1 = 0$, $a_2 = a^2$, and $\hat{s}(n) = s(n)$ is generated by the filter

$$H_1(z) = \frac{1}{1 - a^2 z^{-2}} = \frac{1}{(1 - a z^{-1})(1 + a z^{-1})},$$

excited not by a unit sample, but by the sequence

$$x(n) = u_o(n) + a u_o(n-1).$$

We see that the pole of $H_1(z)$ at $z = -a$ is canceled by the zero at $z = -a$ of the input sequence.

That the predictor coefficients are not, in general, unique is not surprising. The linear-prediction problem as formulated seeks to determine a difference equation, whose solution approximates a given sequence on some interval. If this difference equation is associated with a linear system, we see that there is nothing "built into" the formulation of the problem that specifies the initial conditions of the system, that is, exactly how the system was excited. All that is required by the present formulation of the problem is that the input to the system vanish over the interval on which s(n) is being predicted. Hence the multiplicity of solutions may be interpreted as resulting from the fact that different systems with different inputs can produce identical outputs.

In practice, we are not generally interested in applying linear prediction to a sequence that is exactly the output of an all-pole filter of unknown order. Thus we do not expect the covariance matrix to be singular (when it is singular it can be dealt with by choosing $p = \text{rank } \underline{\Phi}$). We are interested, however, in applying linear prediction to sequences which may be modeled approximately as the output of an all-pole filter. If the sequence

s(n) closely approximates the output of an all-pole filter (as speech often does) the covariance matrix $\underline{\Phi}$ will be ill-conditioned, that is, almost singular. Although a unique solution does exist, it appears to be very sensitive to small perturbations in the data. In particular, if a small amount of noise is added to the data which, according to the previous discussion, results in a family of solutions, the resulting solution may be close to any one of the solutions in the family, and as small perturbations are introduced into the data, the resulting solution may change radically. A consequence is that if the order of the predictor is too large, and the data are close to the unit-sample response of an all-pole filter, as is often assumed to be the case in speech analysis, the unit-sample response of the all-pole filter specified by the linear-prediction parameters and its Fourier transform, may not approximate the data very closely, although the output of the filter resulting from another unspecified input will.

4. Summary and Conclusion

We have attempted to point out the major differences between the various formulations of the linear-prediction problem and discuss a set of important issues related to linear prediction. We have seen that there are generally two different methods of formulating this problem; one requires s(n) to be zero outside the domain of minimization, and the other does not. The autocorrelation method provides a good match to the spectrum, but this is by no means an indication of its superiority over the covariance method.

The uniqueness of the linear-prediction solution is a very important issue. Our experience indicates that with additive noise injected, the system does not always converge to a desirable answer. What perceptual effect this has on speech synthesis is still unclear. We hope to answer this question better after experimental speech synthesis.

M. R. Portnoff, V. W. Zue, A. V. Oppenheim

References

1. B. S. Atal and Suzanne L. Hanauer, "Speech Analysis and Synthesis by Linear Prediction of the Speech Wave," J. Acoust. Soc. Am. 50, 637-665 (1971).

2. C. J. Weinstein and A. V. Oppenheim, "Predictive Coding in a Homomorphic Vocoder," IEEE Trans., Vol. AU-19, pp. 243-248, 1971.

3. J. Markel, "Formant Trajectory Estimation from a Linear Least-Squares Inverse Filter Formulation," Monograph No. 7, Speech Communications Research Laboratory, Inc., Santa Barbara, California, 1971.

4. J. I. Makhoul, "Aspects of Linear Prediction in the Spectral Analysis of Speech," paper presented at IEEE-AFCRL 1972 International Conference on Speech Communication and Processing, Boston, Massachusetts, April 24-26, 1972.

5. V. W. Zue, "Speech Analysis by Linear Prediction," Quarterly Progress Report No. 105, Research Laboratory of Electronics, M.I.T., April 15, 1970, pp. 133-142.

6. N. Levinson, "The Wiener RMS (Root Mean Square) Error Criterion in Filter Design and Prediction," Appendix B in N. Wiener, Extrapolation, Interpolation, and Smoothing of Stationary Time Series (John Wiley and Sons, Inc., New York, 1949).

7. U. Grenander and G. Szegö, Toeplitz Forms and Their Applications (University of California Press, Berkeley, 1958), p. 40.

8. V. N. Faddeeva, Computational Methods of Linear Algebra, English Translation by C. D. Benster (Dover Publications, Inc., New York, 1959), pp. 81-85.

9. J. H. Wilkinson, Rounding Errors in Algebraic Processes (Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1963), pp. 117-118.